

AD-A168 681

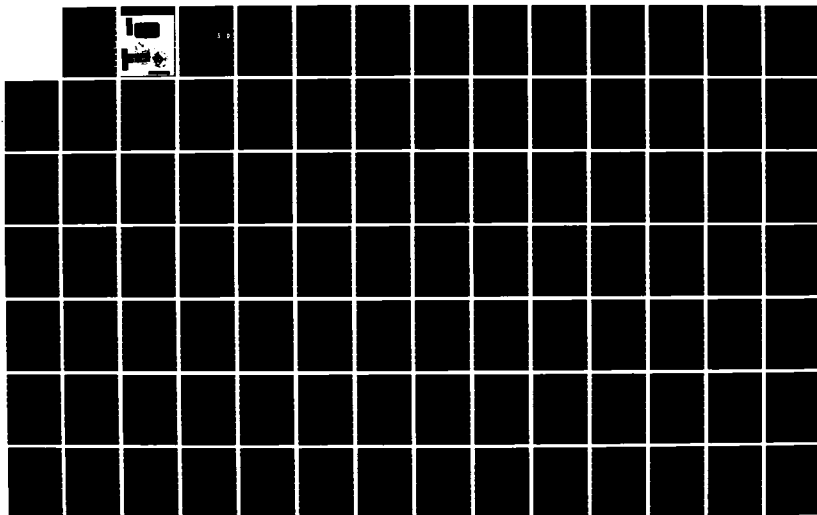
COMPUTER READABILITY AND EDITING SYSTEM (CRES)  
PROGRAMMER'S REFERENCE MANUAL (U) NAVAL TRAINING SYSTEMS  
CENTER ORLANDO FL 5 R SCHALOW ET AL. JAN 86  
NAVTRASYSCEN-TR-86-001

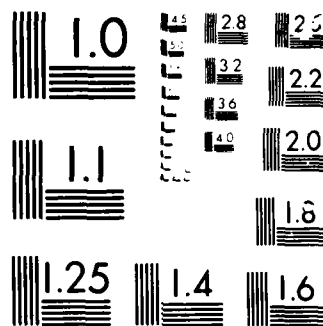
1/4

UNCLASSIFIED

F/G 5/9

NL

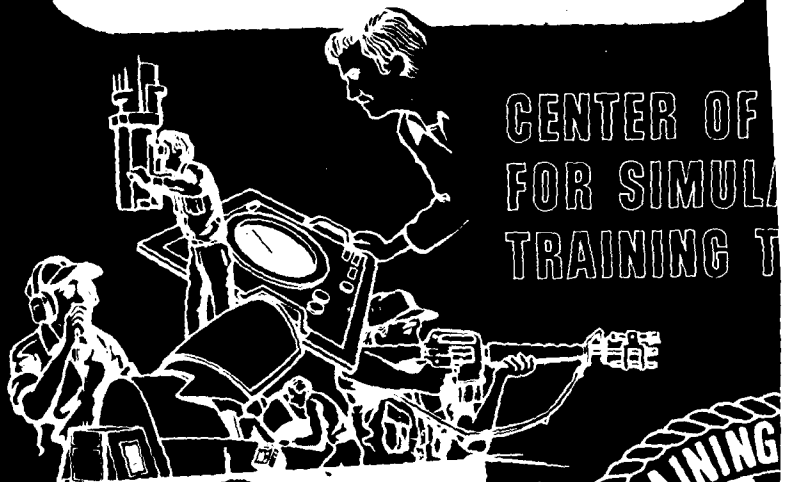




MICROCOPY

CHART

AD-A168



CENTER OF  
FOR SIMULA  
TRAINING T

DTIC  
ELECTE  
JUN 09 1986  
S D



12

TECHNICAL REPORT 86-001  
COMPUTER READABILITY  
AND EDITING SYSTEM  
(CRES)  
PROGRAMMER'S REFERENCE MANUAL

JANUARY 1986

SUSAN R. SCHALOW  
JOHN E. MEARS

DTIC  
ELECTE  
JUN 09 1986  
S D D

TRAINING ANALYSIS AND EVALUATION DEPARTMENT  
NAVAL TRAINING SYSTEMS CENTER  
ORLANDO, FLORIDA 32813-7100

Approved for public release;  
distribution is unlimited.

*T.W. McNaney*  
T.W. McNANEY, Director  
Training Analysis and Evaluation Department

Technical Report 86-001

GOVERNMENT RIGHTS IN DATA STATEMENT

Reproduction of this publication in whole  
or in part is permitted for any purpose  
of the United States Government.

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE

AD-A168681

## REPORT DOCUMENTATION PAGE

1a REPORT SECURITY CLASSIFICATION Unclassified			1b RESTRICTIVE MARKINGS None	
2a SECURITY CLASSIFICATION AUTHORITY			3 DISTRIBUTION/AVAILABILITY OF REPORT Approved for public release; distribution is unlimited	
1d DECLASSIFICATION/DOWNGRADING SCHEDULE				
4 PERFORMING ORGANIZATION REPORT NUMBER(S) Technical Report 86-001			5 MONITORING ORGANIZATION REPORT NUMBER(S)	
6a NAME OF PERFORMING ORGANIZATION Naval Training Systems Center Orlando, FL 32813-7100		6b OFFICE SYMBOL (If applicable) NAVTRASYSSEN	7a NAME OF MONITORING ORGANIZATION	
6c ADDRESS (City, State, and ZIP Code) Naval Training Systems Center (Code 1) Department of the Navy Orlando, FL 32813-7100			7b ADDRESS (City, State, and ZIP Code)	
8a NAME OF FUNDING/SPONSORING ORGANIZATION		8b OFFICE SYMBOL (If applicable)	9 PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER	
8c ADDRESS (City, State, and ZIP Code)			10 SOURCE OF FUNDING NUMBERS	
			PROGRAM ELEMENT NO	PROJECT NO
			TASK NO.	WORK UNIT ACCESSION NO
11 TITLE (Include Security Classification)  COMPUTER READABILITY AND EDITING SYSTEM (CRES) PROGRAMMER'S REFERENCE MANUAL				
12 PERSONAL AUTHOR(S) Susan R. Schalow, John E. Mears				
13a TYPE OF REPORT Final		13b TIME COVERED FROM TO	14 DATE OF REPORT (Year, Month, Day) 1986 January	15 PAGE COUNT 363
16 SUPPLEMENTARY NOTATION				
17 COSATI CODES			18 SUBJECT TERMS (Continue on reverse if necessary and identify by block number) CRES, readability, technical vocabulary, technical word list, common word list, reading grade level, computer text editing	
FIELD	GROUP	SUB-GROUP		
05	07/02			
19 ABSTRACT (Continue on reverse if necessary and identify by block number)  The Navy has long been aware of a literacy gap existing between the level at which Navy technical documents are written and the reading comprehension level of enlisted personnel (GAO 1977 and GAO 1979). In order to provide guidance in the restructuring of text and the limiting of vocabulary to reduce this gap, the Training Analysis and Evaluation Group developed the Computer Readability and Editing System (CRES), reported in TALG Technical Report 83, published in 1980. This original system has been expanded and enhanced to create the current version of CRES. Much emphasis was placed on producing a system which is highly user-oriented and entirely menu-driven. The program software is written in Wang VS BASIC, and is designed to operate on a Wang VS100 system. It can also be transferred to any computer in the Wang VS Series.  This CRES has been distributed to various Navy and Air Force commands requesting the system, as well as to Government contractors (Continued on reverse)				
20 DISTRIBUTION STATEMENT OF ABSTRACT <input checked="" type="checkbox"/> UNCLASSIFIED UNLIMITED <input type="checkbox"/> SAME AS RPT <input type="checkbox"/> OTHERS			21 ABSTRACT SECURITY CLASSIFICATION Unclassified	
22 NAME OF RESPONSIBLE INDIVIDUAL Susan R. Schalow			23 DISTRIBUTION STATEMENT (Include Area Code) (305) 646-4609	
			24 OFFICE SYMBOL NAVTRASYSSEN Code 1A	

DD FORM 1473, 84 MAR

83-10-10-10 may be used if exhausted  
All other editions are obsoleteSECURITY CLASSIFICATION OF THIS PAGE  
Unclassified

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE

19. ABSTRACT (Continued)

having a requirement for it. The purpose of this report is to provide a programmer's reference guide to assist in the implementation, maintenance, and modification of CRES at these facilities. It is also intended to serve as an aid in the conversion of CRES to other languages and machines.

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE

### ACKNOWLEDGEMENTS

Appreciation is expressed to the following people for their support of the development of the Wang VS100 version of the CRES.

J. Peter Kincaid, Ph. D., for his initial conceptualization of this system and his continued interest in its evolution.

Mr. Ray Cox for his development of the original CRES programs.

Ms. Leslie Montgomery, Ms. Rose Tank, and Mr. Bryan Siebuhr for their contributions to the CRES software development.

Ms. Dee Sheppe for her editorial assistance in the preparation of this report.



Accession For	
NTIS	CRA&I <input checked="" type="checkbox"/>
DTIC	TAB <input type="checkbox"/>
Unannounced	<input type="checkbox"/>
Justification	
By	
Distribution	
Availability Codes	
Dist	Avail and/or Special
A-1	

Technical Report 86-001

THIS PAGE INTENTIONALLY LEFT BLANK.

# Technical Report 86-001

## TABLE OF CONTENTS

	<u>Page</u>
INTRODUCTION.....	9
Background.....	9
Purpose.....	9
Organization of the Report.....	11
OVERVIEW OF THE COMPUTER READABILITY AND EDITING SYSTEM.....	13
Subsystems.....	13
Word Lists.....	15
Installation and Operation.....	17
CRES Analysis Subsystem Conventions.....	18
THE COMPUTER READABILITY AND EDITING SYSTEM (CRES).....	19
Master Menu (CRES).....	21
Common Word List Subsystem.....	27
Create/Maintain Common Word List Menu (CRES@CMW).....	27
Create/Maintain Common Word List (CRES@CWL).....	31
Print Common Word List (CRES@CWP).....	45
Print Common Words By Attribute (CRES@CPA).....	55
Technical Word List Subsystem.....	69
Create/Maintain Technical Word List Menu (CRES@SWL).....	69
Create/Maintain Technical Word List (CRES@TWL).....	73
Print Technical Word List (CRES@TWP).....	87
Print Technical Words By Attribute (CRES@TPA).....	99
Word and Phrase Substitute List Subsystem.....	113
Create/Maintain Word and Phrase Substitute List Menu (CRES@WSL).....	113
Create/Maintain Word Substitute List (CRES@WSL).....	117
Print Word Substitute List (CRES@WSP).....	131
Create/Maintain Phrase Substitute List (CRES@PSL).....	141
Print Phrase Substitute List (CRES@PSP).....	153
Print Merged Word and Phrase Substitute List (CRES@SBP).....	163
Glossary Subsystem.....	175
Create/Maintain Glossary Menu (CRES@BCG).....	175
Create/Maintain Abbreviation/Acronym Glossary (CRES@BLG).....	179

# Technical Report 86-001

	<u>Page</u>
Create/Maintain Technical Glossary (CRESGLOS).....	189
Print Abbreviation/Acronym Glossary (CRES@BCP).....	199
Print Technical Glossary (CRESGLOP).....	207
Root Word Expansion Subsystem.....	215
Root Word Expansion (CRES@ERW).....	215
CRES Analysis Subsystem.....	239
User Default Set-up (KIKOFF).....	239
Document and Word List Selection (SCRENZ).....	243
Document Analysis (ANALYSIS).....	251
Title Page Set-up (TTLPGE).....	309
Copy Word Lists To Tape Subsystem.....	313
Copy Word Lists To Tape (CRESTAPE).....	313
REFERENCES.....	319
APPENDIX A WORD LIST DEFINITIONS AND CONTROL FILES.....	323
APPENDIX B LIST OF CRES PROGRAM AND DATA FILES.....	331
APPENDIX C SAMPLE FLOWCHARTS.....	337
CRES Create/Maintain Word List Flowchart.....	341
CRES Print Word List Flowchart.....	343
CRES Print Word List by Attribute Flowchart.....	345
APPENDIX D SAMPLE TEXT PASSAGE AND ANALYSIS.....	347

## LIST OF FIGURES

<u>Figure</u>	<u>Page</u>
1 CRES System Diagram.....	10
2 CRES Overview.....	14
3 CRES Expand Root Words Flowchart.....	216
4 CRES Analysis Flowchart.....	256
5 CRES Copy Word Lists to Tape Flowchart.....	314

## INTRODUCTION

### BACKGROUND

The Navy has the responsibility of providing both general and specialized operation and maintenance training for their personnel. To impart the necessary information the Navy relies heavily on technical documents for training and for on the job maintenance functions. Two studies conducted by the General Accounting Office (GAO 1977 and GAO 1979) indicated much pertinent technical material is written at a higher reading grade level than enlisted personnel can comprehend. In an attempt to deal with this problem, the Chief of Naval Education and Training (CNET), in 1978, tasked the Training Analysis and Evaluation Group (TAEG) to undertake the development of the Computer Readability and Editing System (CRES).<sup>1</sup>

The CRES system analyzes the sentence construction of various reading materials and determines the reading grade level as defined by the Department of Defense Military Specification MIL-M-38784B. In addition, the system provides suggestions for simplification and clarification of the material. The original development and testing of the CRES, along with detailed examples of its use, are described in Development and Test of a Computer Readability Editing System (CRES) (Kincaid, Aagard, and O'Hara, 1980). This original development was done on a Wang 2200MVP computer.

In October 1983 a team was formed to continue this development, expanding the scope and improving the accuracy of the program. The new CRES was to be implemented on a Wang VS100 computer. This effort produced the current CRES; an updated, expanded, and enhanced version written in Wang VS BASIC. Figure 1 identifies key components of CRES and illustrates their interrelations.

### PURPOSE

This report provides a detailed description of the organization of the VS BASIC version of CRES, including the various programs and data files comprising the system. Flowcharts are included for all systems. In the case of three categories of programs the flowcharts are essentially the same for the various CRES subsystems: those to create and maintain word lists, those to print the word lists, and those to print the word lists by specified attribute or combination of attributes. In these cases a single sample flowchart is provided for each genre. The intent of this documentation is to assist programmers in converting CRES from Wang VS BASIC to other computer languages or in the implementation of future modifications.

<sup>1</sup>CNET Code N-5 letter of 29 June 1978.

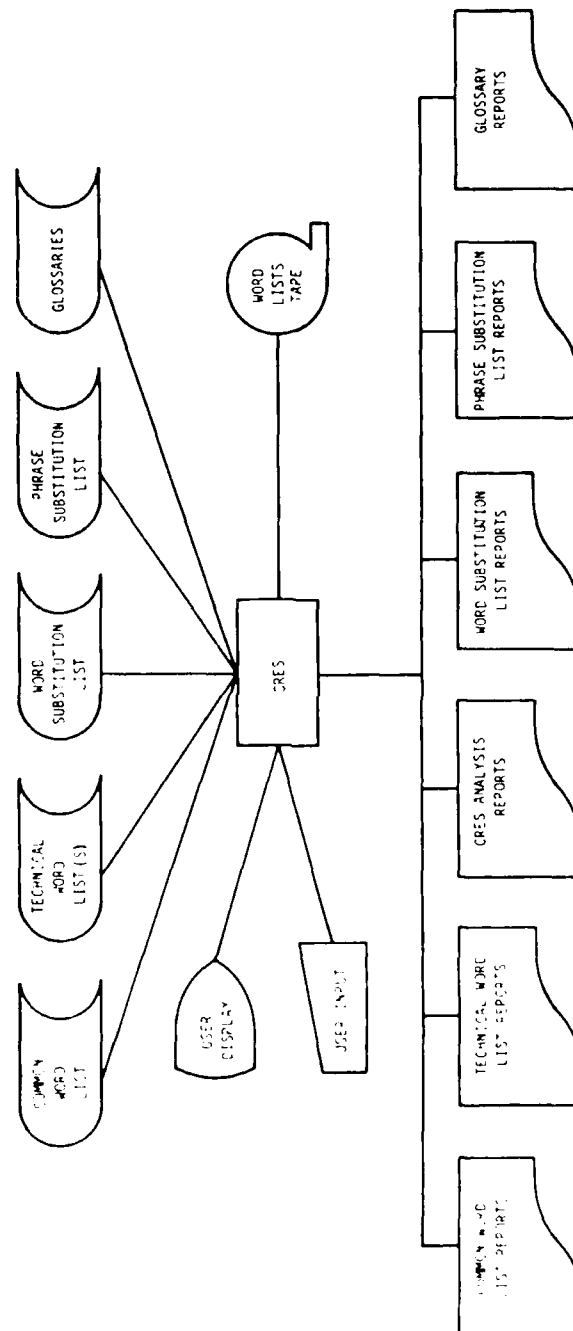


Figure 1. CRES system overview.

## ORGANIZATION OF THE REPORT

In addition to this introduction, the report is divided into two other sections and four appendices. The first section briefly describes the CRES and its major options and suboptions. The next section provides a detailed description of the individual programs comprising the system and a commented listing for each program. Appendix A contains the word list definitions and the control file describing each word list data file. Appendix B contains a list of all the programs and data files included in the CRES, along with their library and volume locations. Appendix C contains sample flowcharts for the programs to create and maintain the word lists, to print the word lists, and to print the word lists by a specified attribute or combination of attributes. Appendix D contains a sample text passage and the output obtained from a CRES analysis of the text.

Technical Report 86-001

THIS PAGE INTENTIONALLY LEFT BLANK.

## OVERVIEW OF THE COMPUTER READABILITY AND EDITING SYSTEM

The CRES supports the generation of a readability analysis for a given text, the generation of editorial suggestions for simplifying and clarifying the text, the creation of a glossary of technical terms and their definitions for reference purposes, and the creation and maintenance of word and phrase lists to be used in the CRES analysis. Further information on the usage of CRES can be found in Computer Readability Editing System (CRES) for Writers and Editors (Kincaid, Pierce, in press).

### SUBSYSTEMS

Figure 2 depicts the organization of the CRES and the interconnection of the various subsystems which is achieved through a chain of menus. The system is comprised of seven distinct subsystems:

- 1 - The Common Word List Subsystem
- 2 - The Technical Word List Subsystem
- 3 - The Word and Phrase Substitute Subsystem
- 4 - The Glossary Subsystem
- 5 - The Root Word Expansion Subsystem
- 6 - The Analysis Subsystem
- 7 - The Copy Word Lists to Tape Subsystem

The Common Word List Subsystem enables the user to create, modify, or print out the Common Word List. The Common Word List is a compilation of words which are considered familiar and which frequently occur in ordinary usage. These words are defined as "common". When text is analyzed, each word is checked against this list; words not appearing on the Common Word List are considered "uncommon" words and will be flagged as such. At the present time, the Common Word List is comprised of approximately 4000 root words and their expansions, yielding approximately 11,000 words. A list of all words contained in the Common Word List can be found in Word Lists to Limit Vocabulary in Technical Information (Kincaid, McDaniel, Pierce, 1985).

The Technical Word List Subsystem enables the user to create, modify, or print out the various technical word lists. A technical word list enables a user to specify certain technical words which, although uncommon in ordinary usage, are commonly encountered in a particular area. They should be considered common in any document pertaining to that area. In conducting a CRES analysis, the user is given the opportunity to specify up to seven different technical word lists. At the present time, the system includes Administrative/Clerical; Aircraft Equipment; Communication/Navigation; General Electronics; Hull, Mechanical, and Electrical; Propulsion Engineering; and Weapon System Control Technical Word Lists. Any or all of these, or any

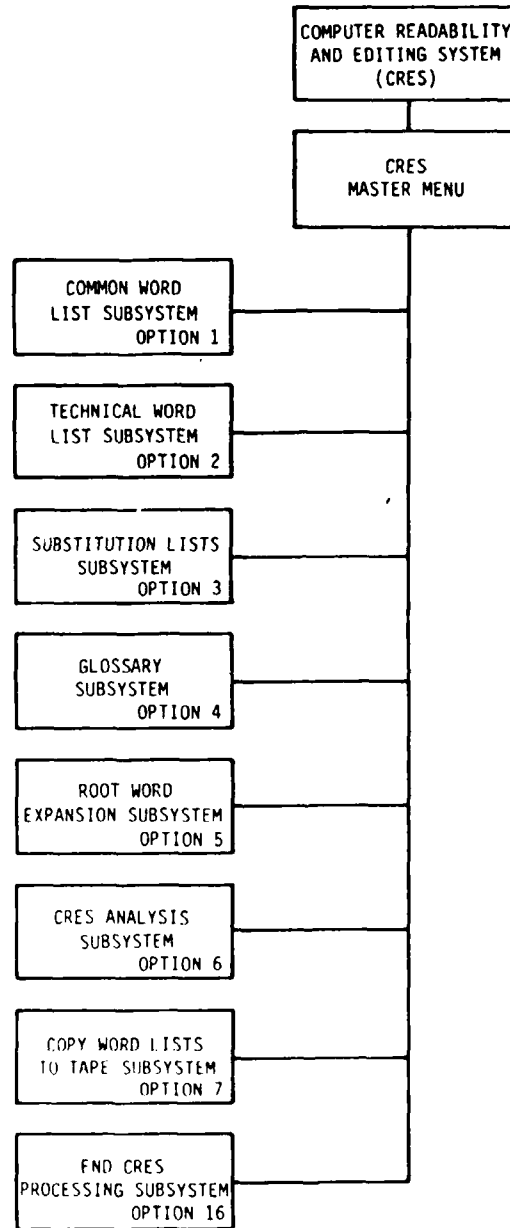


Figure 2. CRES system diagram.

## Technical Report 86-001

additional list(s) created by the user through the Technical Word List Subsystem, may be specified for inclusion in an analysis. Any lists specified are considered as supplements to the Common Word List for the purposes of that analysis, and any words appearing on them will be considered to be "common". A copy of each of the technical word lists has been published (Kincaid, et al, 1985).

The Word and Phrase Substitute Subsystem enables the user to create, modify, or print out the Word Substitution List or the Phrase Substitution List. These lists provide up to two suggested substitutes for the given word or phrase. If requested by the user, these suggested substitutes will be provided in the CRES analysis.

The Glossary Subsystem enables the user to create, modify, or print out any glossaries desired. There are two different types of glossaries treated in this system, which are useful adjuncts to the text when writing technical material. The first is an Abbreviation-Acronym Glossary, which contains various abbreviations, acronyms, and their definitions. This glossary is consulted by the Analysis Subsystem. The second is a Technical Glossary, which is comprised of various technical terms and their definitions. This is included exclusively for the user's reference and is not utilized by any other part of the CRES.

The Root Word Expansion Subsystem enables the user to expand any root word as required and to merge these expansions into the Common Word List or into a technical word list.

The Analysis Subsystem enables the user to perform a CRES analysis of any VS word processing document. The user has the capability to specify various options, including calculation of the reading grade level and the flagging of various characteristics which raise the reading grade level.

The Copy Word Lists to Tape Subsystem enables the user to transfer to tape any or all of the CRES word lists.

### WORD LISTS

The CRES Analysis Subsystem depends on many word lists. Appendix A contains the data file definition and the control file for each of these lists. The following is a brief discussion of the different types of word lists, and the naming conventions used:

CMNWDLST : This is the Common Word List. This list contains the words considered to be "common" words, (familiar and frequently occurring), with their parts of speech and other pertinent information. There are approximately 11,000 entries in the current list.

## Technical Report 86-001

- WRDSBLST : This is the Word Substitution List. This list consists of words considered to be awkward or unnecessarily difficult, with one or two suggested substitutes for each word. A substitute can be any word or phrase of 20 characters or less.
- PHRSBLST : This is the Phrase Substitution List. This list consists of phrases considered to be awkward or unnecessarily difficult, with one or two suggested substitutes for each phrase. A substitute can be any word or phrase of 24 characters or less.
- ABBRELST : This is a list of common abbreviations and acronyms, and their definitions.
- GLOSSARY : This is an optional file used to contain any desired words and their definitions. It can be printed out for reference, but it is not utilized in the analysis.

In the development of CRES, there were seven different technical word lists established for testing and evaluation. Each technical word list contains words that, while they might be unusual to a novice, represent common terminology to someone more familiar with the subject area. These words would not be flagged as uncommon if the technical word list containing them is specified for use in the analysis. The lists provided are as follows:

- ADMCLER : Administrative and Clerical Word List (291 root words, 607 total words)
- AIPEQUI : Aircraft Equipment Word List (189 root words, 367 total words)
- COMMNAV : Communications and Navigation Word List (413 root words, 770 total words)
- GENELECT : General Electronics Word List (359 root words, 669 total words)
- HMELECT : Ships' Mechanical and Electrical Word List (471 root words, 919 total words)
- PROPENG : Propulsion Engineering Word List (443 root words, 885 total words)
- WSCONT : Weapon Systems Control Word List (375 root words, 885 total words)

## Technical Report 86-001

The CRES system supports the maintenance of existing lists, and the creation of new lists where required. Each installation is encouraged to develop its own word lists, as well as to modify and use the seven lists provided.

### INSTALLATION AND OPERATION

The CRES is designed to be implemented on the Wang VS system, running on Wang Operating System Release 5.03.10A or later. It requires a standard VS configuration, including a workstation, line printer, and disk drive with at least 5MB of disk storage.

In order to run the CRES, certain Wang utilities are required. The object code for these programs must be placed in library @SYSSAV@ on the volume where the CRES system resides. The following is a list of the required utilities:

DISMOUNT  
DISPLAY  
EXTRACT  
LINK  
MOUNT  
READFOR  
READVTOC  
SCRATCH

In addition, the object code for the word processing subroutine program, WPSBRTNB, must be placed in library @LINKLIB on the volume where CRES resides.

At the time of installation, it is necessary for a programmer to provide the name of the volume where the CRES will reside. The following steps are required:

- 1 - Edit program CRES in library CRES@PRC on the desired volume name.
- 2 - On line number 2000, the variable &CARVOL is currently initialized to CRES00. This initial value must be changed to the desired volume name for the CRES volume.
- 3 - After this change has been made, the input file (filename CRES in library CRES@PRC on the CRES volume) must be replaced with the edited file.

To operate the CRES, run filename CRES in library CRES@PRC on the volume on which CRES resides. This produces the CRES Master Menu, from which any subsystem can be selected. (See Figure 2.) It is advisable to establish a logon procedure to run CRES, and to assign this logon procedure to those users who will be running the system.

## Technical Report 86-001

Detailed instructions for the operation of the CRES can be found in the Computer Readability and Editing System (CRES): User's Reference Manual (S. R. Schalow and J. E. Mears, in press).

### CRES ANALYSIS SUBSYSTEM CONVENTIONS

The CRES Analysis Subsystem will perform a CRES analysis on any text that is in word processing document format. Any particular portion of text in the document to be excluded from the analysis must be surrounded by the characters "@@". An "@@" sign must immediately precede the text to be excluded, and another "@@" sign must immediately follow, signaling the analysis to resume.

The Analysis Subsystem supports a defaults file for each user. This defaults file holds the names of technical word lists to be used (displayed on the Technical File Selection Screen) and analysis options selected (displayed on the Option Selection Screen) for each user. After the defaults file is established for a particular user, the analysis program is initialized to that user's defaults on subsequent analyses. If the defaults are not satisfactory, the user can change them for any particular session when running an analysis. If a defaults file does not already exist for a given user, the Analysis Subsystem is designed to create one, which will contain the options selected by the user for the first CRES analysis performed. The filename is formed by the User ID concatenated with the string "DFLTS", and is stored in library CRES@DAT. The user can permanently change the defaults by scratching the defaults file. The program will then recreate the file in the next analysis run, using the new options selected for future defaults.

The system can support multiple users although, if multiple analyses are being run, each analysis must be examining a different document. To avoid the possibility of more than one user accessing a word simultaneously during maintenance of the word lists, a protection system has been implemented. Protection bytes are set in the record of a word when it is called up for editing. If another user tries to access the word while the protect bytes are set, a message will be presented to this effect. When the editing of the word has been completed, the bytes are reset to zero. This process is ordinarily transparent to the user. However, in the case of a system crash or a power failure, it would be possible for the protect bytes to remain set. Should this happen, capability has been provided to reset the protect bytes by use of the appropriate option on the editing subsystem menu.

The following section of this manual provides the detailed descriptions and listings of the programs in the system.

## THE COMPUTER READABILITY AND EDITING SYSTEM (CRES)

This section contains the listings, including comments, for the programs comprising the CRES and necessary explanatory notes. Flowcharts are included with specific programs, except in the case of programs to create and maintain a word list, to print a word list, or to print a word list by attribute. Sample flowcharts for these three types of programs are included in appendix C.

The programs are written primarily in VS BASIC, with the exception of the menus. Those are written in VS Procedure Language. A number of the CRES programs have occasion to call Wang VS utility programs, such as "DISMOUNT", "DISPLAY", "EXTRACT", "LINK", "MOUNT", "READFDR", "READVTOC", and "SCRATCH". The code for these utilities is only available from Wang Laboratories, Incorporated, and no discussion of them can be provided in this document.

The programs contained in this section are organized in the same order as in the Master Menu.

Technical Report 86-001

THIS PAGE INTENTIONALLY LEFT BLANK.

## Technical Report 86-001

### MASTER MENU (CRES)

CRES in library CRES@PRC is the Procedure Language menu routine that serves as the driver for the entire CRES system. It enables the user to select by PF key among options to enter the Common Word List Subsystem, the Technical Word List Subsystem, the Substitution Lists Subsystem, the Glossary Subsystem, the Analysis Subsystem, and the Copy Word Lists to Tape Subsystem, or to end CRES processing.

A complete listing follows.

Technical Report 86-001

THIS PAGE INTENTIONALLY LEFT BLANK.

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 15:59 09/19/84 Page 1  
Input File is CRES in Library CRES@PRC on Volume CRES00

```

*
*      PROCEDURE      CRES MASTER MENU
*      AUTHOR        ROSE TANK
*      DATE          10/06/83
*
*      Abstract: This procedure is the CRES MASTER MENU .
*                It allows the user to select and run any
*                file option.
*
*      DECLARE        &RNDOM   AS STRING (1) INITIAL " "
*      DECLARE        &USERINPT AS INTEGER   INITIAL 0
*      DECLARE        &OPTIONERR AS STRING (79) INITIAL " "
*      DECLARE        &OPTIONERR2 AS STRING (79) INITIAL " "
*      DECLARE        &USERNAME AS STRING (24) INITIAL " "
*      DECLARE        &ERRMSG   AS STRING (34) INITIAL " "
*      DECLARE        &RUNLIB   AS STRING (8) INITIAL " "
*      DECLARE        &RUNVOL   AS STRING (6) INITIAL " "
*      DECLARE        &SUNLIB   AS STRING (8) INITIAL " "
*      DECLARE        &SUNVOL   AS STRING (6) INITIAL " "
*
*      * &CARVOL IS THE VARIABLE THAT NEEDS TO BE ALTERED IN LINE 2000.
*      * REPLACE "CRES00" WITH THE NAME OF THE VOLUME AT YOUR INSTALLATION.
*      * DON'T FORGET TO REPLACE THE INPUT FILE (CRES IN CRES@PRC ON YOUR
*      * VOLUME), WHICH NOW HAS YOUR VOLUME NAME INITIALLY.
*
*      DECLARE        &CARVOL AS STRING (6) INITIAL "CRES00"
*
*      DECLARE        &SINVOL AS STRING (6) INITIAL " "
*      DECLARE        &SINLIB AS STRING (8) INITIAL " "
*      DECLARE        &SOTVOL AS STRING (6) INITIAL " "
*      DECLARE        &SOTLIB AS STRING (8) INITIAL " "
*      DECLARE        &SRNLIB AS STRING (8) INITIAL " "
*      DECLARE        &SRNVOL AS STRING (6) INITIAL " "
*      DECLARE        &WRKVOL AS STRING (6) INITIAL " "
*      DECLARE        &SPOLIB AS STRING (8) INITIAL " "
*      DECLARE        &SPOVOL AS STRING (6) INITIAL " "
*
*      EXTRACT &SRNLIB = RUNLIB, &SRNVOL=RUNVOL, &SINLIB=INLIB,
*              &SINVOL = INVOL, &SOTVOL = OUTVOL, &SOTLIB=OUTLIB,
*              &WRKVOL = WORKVOL, &SPOLIB = SPOOLIB, &SPOVOL = SPOOLVOL
*
*      SET      INLIB=CRES@DAT,      INVOL=&CARVOL,      OUTLIB=CRES@SAV,
*              OUTVOL=&CARVOL,      RUNLIB=CRES@PRC,      RUNVOL=&CARVOL,
*              WORKVOL=&CARVOL,      SPOOLVOL=&CARVOL,      SPOOLIB=CRES@PRT
*
*      EXTRACT &USERNAME = USERNAME, &RUNLIB = RUNLIB, &RUNVOL=RUNVOL,
*              &SUNLIB = OUTLIB, &SUNVOL = OUTVOL
*
*      GETUSERINPT:  PROMPT PFKEY = &USERINPT  CENTER
*                    DIM "CRES: Computer Readability & Editing System ","
*                    (Master Menu Screen)";
*                    DIM "=====
*                    =====";
*                    DIM " ";
*                    DIM " Hello,",bright &USERNAME(1,*),".", " Welcome to the CRES Ma

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 15:59 09/18/84 Page 2  
Input File is CRES in Library CRES@PRC on Volume CRES00

```

ster Menu, your guide through ";
    DIM " the Computer Readability & Editing System.";
    DIM " Select the", BRIGHT " PF key", " corresponding to your desir
ed operation.";
    DIM " ";
    DIM " ",LINE "PF",DIM " ",
    LINE " Operation ";
    DIM " ";
    DIM " ",Bright "1", " - Create/Maintain Common Word
List";
    DIM " ",Bright "2", " - Create/Maintain Technical W
ord List";
    DIM " ",Bright "3", " - Create/Maintain Substitutio
n Lists";
    DIM " ",Bright "4", " - Maintain Glossaries";
    DIM " ",Bright "5", " - Expand Root Words";
    DIM " ",Bright "6", " - Analyze Word Processing Doc
uments";
    DIM " ",Bright "7", " - Copy Word Lists to Tape";
    DIM " ";
    DIM " ",Bright "16", " - End CRES Processing";
    BRIGHT &OPTIONERR;
    BLINK &OPTIONERR2;
*
    ASSIGN &ERRMSG = " "
    ASSIGN &OPTIONERR = " "
    ASSIGN &OPTIONERR2 = " "
*
    IF &USERINPT EQ 16 GOTO USERLOG
    IF &USERINPT EQ 32 GOTO USERLOG
    IF &USERINPT EQ 1 GOTO RUNCMW
    IF &USERINPT EQ 2 GOTO RUNSWL
    IF &USERINPT EQ 3 GOTO RUNWSL
    IF &USERINPT EQ 4 GOTO RUNBCG
    IF &USERINPT EQ 5 GOTO RUNERW
    IF &USERINPT EQ 6 GOTO RUNATF
    IF &USERINPT EQ 7 GOTO RUNCTP
    IF &USERINPT EQ 17 GOTO RUNCMW
    IF &USERINPT EQ 18 GOTO RUNSWL
    IF &USERINPT EQ 19 GOTO RUNWSL
    IF &USERINPT EQ 20 GOTO RUNBCG
    IF &USERINPT EQ 21 GOTO RUNERW
    IF &USERINPT EQ 22 GOTO RUNATF
    IF &USERINPT EQ 23 GOTO RUNCTP
    ASSIGN &OPTIONERR "I'm sorry, but your selected option
is not currently available."
    ASSIGN &OPTIONERR2 = "Please select another topic."
    GOTO GETUSERINPT
*
    RUNCMW: RUN CRES@CMW IN &RUNLIB ON &RUNVOL
    IF RUNCMW EQ 0 GOTO GETUSERINPT
    GOTO USEREXIT
*
    RUNSWL: RUN CRES@SWL IN &RUNLIB ON &RUNVOL
    IF RUNSWL EQ 0 GOTO GETUSERINPT
    GOTO USEREXIT

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 15:59 09/18/84 Page 3  
Input File is CRES in Library CRES@PRC on Volume CRES00

*		BP	010900
RUNWSL:	RUN CRES@WSL IN &SUNLIB ON &RUNVOL	RT	011000
	IF RUNWSL EQ 0 GOTO GETUSERINPT	RT	011100
	GOTO USEREXIT	BP	011200
*		BP	011300
RUNERW:	RUN CRES@ERW IN &SUNLIB ON &SUNVOL	RT	011400
	IF RUNERW EQ 0 GOTO GETUSERINPT	RT	011500
	GOTO USEREXIT	RT	011600
*		RT	011700
RUNATF:	RUN ANALYSIS IN &SUNLIB ON &SUNVOL	RT	011800
	IF RUNATF EQ 0 GOTO GETUSERINPT	RT	011900
	GOTO USEREXIT	BP	012000
*		RT	012100
RUNCTP:	RUN CRESTAPE IN &SUNLIB ON &SUNVOL	RT	012200
	IF RUNCTP EQ 0 GOTO GETUSERINPT	RT	012300
	GOTO USEREXIT	RT	012400
*		RT	012500
RUNBCG:	RUN CRES@BCG IN &SUNLIB ON &RUNVOL	RT	012600
	IF RUNBCG EQ 0 GOTO GETUSERINPT	RT	012700
	GOTO USEREXIT	BP	012800
*		RT	012900
RUNMTG:	RUN CRESTAPE IN &SUNLIB ON &SUNVOL	RT	013000
	IF RUNMTG EQ 0 GOTO GETUSERINPT	RT	013100
	GOTO USEREXIT		013200
*		RT	013300
USEREXIT:	RETURN	BP	013400
*		RT	013500
USERLOG: SET	INLIB=&SINLIB, INVOL=&SINVOL, OUTLIB=&SOUTLIB,	RT	013600
	OUTVOL=&SOTVOL, RUNLIB=&SRNLIB, RUNVOL=&SRNVOL,	RT	013700
	WORKVOL=&WRKVOL, SPOOLVOL=&SPOVOL, SPOOLIB=&SPOOLIB	JM	013800
*		JM	013900
USERLOGOFF:	RUN CRES@RND IN &SUNLIB ON &SUNVOL	RT	014000
	IF USERLOGOFF EQ 1 GOTO RUNENA		014100
	IF USERLOGOFF EQ 2 GOTO RUNENB		014200
	IF USERLOGOFF EQ 3 GOTO RUNENC		014300
	IF USERLOGOFF EQ 4 GOTO RUNEND		014400
	IF USERLOGOFF EQ 5 GOTO RUNENE		014500
	IF USERLOGOFF EQ 6 GOTO RUNENF		014600
	IF USERLOGOFF EQ 7 GOTO RUNENG		014700
	GOTO USEREND	RT	014800
*		RT	014900
RUNENA:	RUN CRSEN1 IN &SUNLIB ON &SUNVOL	RT	015000
	GOTO USEREND		015100
*		RT	015200
RUNENB:	RUN CRSEN2 IN &SUNLIB ON &SUNVOL	RT	015300
	GOTO USEREND		015400
*		RT	015500
RUNENC:	RUN CRSEN3 IN &SUNLIB ON &SUNVOL	RT	015600
	GOTO USEREND		015700
*		RT	015800
RUNEND:	RUN CRSEN4 IN &SUNLIB ON &SUNVOL	RT	015900
	GOTO USEREND		016000
*			016100
RUNENE:	RUN CRSEN5 IN &SUNLIB ON &SUNVOL	RT	016200
	GOTO USEREND		016300
*		RT	016400
RUNENF:	RUN CRSEN6 IN &SUNLIB ON &SUNVOL	RT	016500

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 15:59 09/18/84 Page 4  
Input File is CRES in Library CRES@PRC on Volume CRES00

* GOTO USEREND	RT 016600
* RUNENG: RUN CRSEN7 IN &SUNLIB ON &SUNVOL	RT 016700
* GOTO USEREND	RT 016800
* USEREND:	016900
*	017000
*	017100
*	RT 017200
*	017300
* RETURN CODE = 16	017400

## COMMON WORD LIST SUBSYSTEM

### Create/Maintain Common Word List Menu (CRES@CMW)

CRES@CMW in library CRES@PRC is the Procedure Language menu routine that serves as the driver for the Create/Maintain Common Word List Subsystem. It enables the user to select by PF key among options to create/maintain the Common Word List, to print the Common Word List, to print the Common Word List by attribute, or to return to the CRES Master Menu.

A complete listing follows.

Technical Report 86-001

THIS PAGE INTENTIONALLY LEFT BLANK.

## COMMON WORD LIST SUBSYSTEM

### Create/Maintain Common Word List Menu (CRES@CMW)

CRES@CMW in library CRES@PRC is the Procedure Language menu routine that serves as the driver for the Create/Maintain Common Word List Subsystem. It enables the user to select by PF key among options to create/maintain the Common Word List, to print the Common Word List, to print the Common Word List by attribute, or to return to the CRES Master Menu.

A complete listing follows.

Technical Report 86-001

THIS PAGE INTENTIONALLY LEFT BLANK

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 15:54 11/13/84 Page 1  
Input File is CRES@CMW in Library CRES@PRC on Volume CRES01

```

*
*      PROCEDURE      CRES COMMON WORD LIST MENU
*      AUTHOR        ROSE PANK
*      DATE          10/06/83
*
*      Abstract: This procedure is a subsystem of CRES Master Menu
*                (option 1 - Create/Maintain Common Word List).
*                It allows the user to select and run any Common
*                word list file option.
*
*      DECLARE        &USERINPT AS INTEGER      INITIAL 0
*      DECLARE        &OPTIONERR AS STRING (79) INITIAL "
*      DECLARE        &OPTIONERR2 AS STRING (79) INITIAL "
*      DECLARE        &USERNAME AS STRING (24) INITIAL "
*      DECLARE        &ERRMSG AS STRING (34) INITIAL "
*      DECLARE        &RUNLIB AS STRING (8) INITIAL "
*      DECLARE        &RUNVOL AS STRING (6) INITIAL "
*
*      EXTRACT &USERNAME=&USERNAME, &RUNVOL=&RUNVOL, &RUNLIB=&OUTLIB
*
*      GETUSERINPT:  PROMPT PFKEY = &USERINPT      CENTER
*                    DIM "CRES: Common Word List Subsystem      (Common Word
*                      List Menu Screen)";
*                    DIM "=====
*                    DIM " ";
*                    DIM " The Common Word Lists will be used to perform the CRES ana
*      lysis.";
*                    DIM " Use the appropriate",BRIGHT " PF key"," to select the desi
*      red operation.";
*                    DIM " ";
*                    DIM "          ",LINE "PF",DIM " ",
*                    LINE          Operation
*                    ";
*                    DIM " ";
*                    DIM "          ",BRIGHT "1"," - Create/Maintain Common Word
*      List";
*                    DIM "          ",Bright "2"," - Print Common Word List";
*                    DIM "          ",Bright "3"," - Print Common Words by Attri
*      bute";
*                    DIM " ";
*                    DIM "          ",Bright "16"," - Return to CRES Master Menu";
*                    DIM " ";
*                    DIM " ";
*                    BRIGHT &OPTIONERR;
*                    BLINK &OPTIONERR2;
*
*      ASSIGN        &ERRMSG = " "
*      ASSIGN        &OPTIONERR = " "
*      ASSIGN        &OPTIONERR2 = " "
*
*      IF &USERINPT EQ 16      GOTO USEREXIT
*      IF &USERINPT EQ 32      GOTO USEREXIT
*      IF &USERINPT EQ 1       GOTO RUNCWL
*      IF &USERINPT EQ 2       GOTO RUNPWL
*      IF &USERINPT EQ 3       GOTO RUNPWA
*      IF &USERINPT EQ 17      GOTO RUNCWL
*      IF &USERINPT EQ 18      GOTO RUNPWL

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 15:54 11/13/84 Page 2  
Input File is CRES@CWL in Library CRES@PRC on Volume CRES01

```

      IF      &USERINPT EQ 19                      GOTO RUNPWA          RT 005935
                                              RT 005965
      ASSIGN &OPTIONERR = "I'm sorry, but your selected option
is not currently available." RT 006000
      ASSIGN &OPTIONERR2 = "Please select another topic." RT 006100
      GOTO GETUSERINPT RT 006200
* RT 006300
* RT 006400
RUNCWL: RUN CRES@CWL IN &RUNLIB ON &RUNVOL RT 006500
      IF RUNCWL EQ 0 GOTO GETUSERINPT RT 006600
      GOTO USEREXIT RT 006700
* RT 006800
* RT 007300
RUNPWL: RUN CRES@CWP IN &RUNLIB ON &RUNVOL RT 007400
      IF RUNPWL EQ 0 GOTO GETUSERINPT RT 007500
      GOTO USEREXIT RT 007600
* RT 007600
* RT 008500
RUNPWA: RUN CRES@CPA IN &RUNLIB ON &RUNVOL RT 008600
      IF RUNPWA EQ 0 GOTO GETUSERINPT RT 008700
      GOTO USEREXIT RT 008800
* RT 008900
USEREXIT: RETURN

```

### Create/Maintain Common Word List (CRES@CWL)

CRES@CWL in library CRES@SRC is the BASIC language program to create/maintain the Common Word List. The object language version is CRES@CWL in library CRES@SAV.

This program first allows the user to specify the word list to be created or maintained. Then the program enables the user to add, delete, or modify any given word. The program also provides for a protection override. The protection override option allows the user to clear the protect bytes, if for some reason these bytes have been left set incorrectly.

The protection capability was included to accommodate multiple users on the system. When a word on any of the CRES word lists is being modified by one user, the protection bytes for the word are set to that user's ID. No other user can access that word while the protection bytes are set. If an attempt is made by one user to access a word protected by another user, a message specifying the user and workstation protecting the word is displayed. When the word is released, the protection bytes are automatically cleared. To provide for the possibility of the system going down, a power failure, or an irregular exit leaving a word with its protection bytes still set, the program contains a protection override capability. For a given user to override the protection, either the word had to be protected under that user's ID, or the user protecting the word has to be no longer logged on to the computer.

The only characters accepted in the input word are the letters A through Z, ".", "-", and "'". There can be no embedded blanks. The data record for each word also contains information about the word's attributes: the part of speech (see table 1), the past participle flag, the negative flag, the form of the verb "to be" flag, and the root word flag<sup>1</sup>. The complete data record description is provided in appendix A.

A complete program listing follows. Appendix C contains a sample flowchart.

<sup>1</sup>A root word is defined, for the purposes of this program, as the form of a word which cannot be reduced any further; i.e., the singular, present tense word for nouns, verbs and adjectives. Most other parts of speech have only one form, which is considered the root word.

Table 1  
Parts of Speech

<u>PART OF SPEECH</u>	<u>DEFINITION</u>
(J) Adjective .....	Modifies a noun or pronoun (e.g. pretty, little)
(j) Adjective/verb.....	A word that may function as an adjective or verb (e.g. aged, dry, lean, open)
(A) Adverb.....	Modifies a verb, adjective, or adverb (e.g. quickly, highly, very)
(a) Adverb/verb.....	A word that may function as an adverb or verb (e. g. close, near, slow)
(B) Article.....	A limiting adjective before a noun (e.g. a, the)
(C) Conjunction.....	Links phrases or clauses (e.g. or)
(I) Interjection.....	Used for simple exclamations (e.g. Oh!)
(N) Noun.....	Indicates a person, place, or thing (e.g. woman, city, box)
(d) Noun/adjective.....	A word that may function as a noun or adjective (e.g. metric, primitive)
(n) Noun/verb.....	Word that may fuction as a noun or verb (e.g. abuse, dash, spot, wind)
(P) Preposition.....	Expresses a relationship between two elements of a sentence (e.g. in, on, by, to)
(O) Pronoun.....	Word that may be used as a substitute for a noun (e.g. I, you, she, what)
(V) Verb.....	Denotes an action or occurrence (e.g. run, yawn, happen)

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 11:35 11/09/84 Page 1  
Input File is CRESECW1 in Library CRES\$SRC on Volume CRES02

```

000010 /*      This is the CRES COMMON WORD LIST MAINTENANCE PROGRAM      */ JM
000060 /*                                                    */ RT000020
000110 /*      AUTHORS: Rose Tank and John Mears                        */ RT000030
000160 /*      Date:      October 12, 1983                               */ RT000040
000210 /*                                                    */ RT000050
000260 /*                                                    */ JM
000310 /* This program was developed to maintain the common word list. */ RT
000360 /* It is designed to insert, delete, and modify any word in a */ RT
000410 /* common word list (in the old CRES system, it was referred to */ JM
000460 /* as the Master Word List). This subsystem has also been ex- */ RT
000510 /* panded to handle multiusers by use of protection bytes set */ JM
000560 /* in the record containing the word to be modified. There is */ RT
000610 /* an override capability within the program itself that */ JM
000660 /* operates on the condition that the user's id, which is ex- */ JM
000710 /* tracted from the system, matches the protection bytes. In */ JM
000760 /* the case where the user attempts to modify a word currently */ RT
000810 /* controlled by another, a message is given identifying the */ JM
000860 /* user and his/her workstation number.                        */ JM
000910 /*                                                    */ RT
000960 /* The record layout for the common word list is as follows: */ RT
001010 /*                                                    */ RT
001060 /* character position      identifier      variable name */ RT
001110 /*          1-20          => word          WORD$          */ RT
001160 /*          21            => part of speech  PRTOPSPC$      */ RT
001210 /*          22            => past participle PPARTFL$       */ RT
001260 /*          23            => negative        NEGFL$         */ RT
001310 /*          24            => to be flag      TOBEFL$         */ RT
001360 /*          25            => filler          */ RT
001410 /*          26            => root word       ROOTFL$         */ RT
001420 /*          27-28         => filler          */ JM
001510 /*          29-31         => protection bytes PROTECT$      */ JM
001560 /*          32-40         => filler for future additions */ RT
001610 /*                                                    */ RT
001660 /* The words are stored in an indexed file with the word */ JM
001710 /* acting as the key. */ JM
001760 RT000120
001810 RT000130
001860 DIM WORD$20,IOBUFF$40,PRTOPSPC$1,PPARTFL$1,NEGFL$1,MWLFL$1 RT
001910 DIM TOBEFL$1,ROOTFL$1,PROTECT$3,PF$20,F2$10,MESING$14 JM
001960 DIM USER$3,USERNAM$24,FILE$8,LIB$8,IVOL$6 RT
002010 DIM CODE$1,ERL$50,ITLE$21,ZAP$23,ER$65,X$1 JM
002060 DIM ES$1,GS$1,L$1,V$1,B$1,C$1,D$1,PASSWRD$11 JM
002110 DIM PF1$3,PF16$4,ENTER$5,HEADER$43,PF$4,OPER$12,TAB$3,BTAB$7 JM
002160 DIM MOD$15,EXP$30,DEF$7,AS$10,LEG$5,A7$7,PS(13)3,ITLE5$14 JM
002165 DIM ITLE6$10,FIL$16 JM
002210 HEADER$ = "CRES COMMON WORD LIST MAINTENANCE SUBSYSTEM" JM
002260 FIL$ = "Common Word List" RT
002310 PF16$ = "PF16" RT
002360 TAB$ = "TAB" RT000340
002410 BTAB$ = "BACKTAB" RT000350
002460 EXP$ = "(Definition of Entries Screen)" JM
002510 ITLE$ = "Parts of Speech Codes" JM
002560 ENTER$ = "ENTER" RT
002610 WORDEQ$ = "word = " 001230
002660 PF$ = " PF " RT
002710 OPER$ = " Operation" 001250
002760 QS = "1":TS = "2":TH$ = "3":QS = "14":SS = "16" JM

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 11:35 11/09/84 Page 2  
Input File is CRES@CWL in Library CRES@SRC on Volume CRES02

```

002810 PF1$          ="PF1"          RT
002860 DEF$          ="CURRENT"      RT002760
002910 A5$          ="ATTRIBUTES"    RT002770
002960 LEG$         ="LEGAL"         RT002780
003010 A7$          ="ENTRIES"       RT002790
003015 P$(1)="(J)":P$(2)="(j)":P$(3)="(A)":P$(4)="(a)":P$(5)="(B)" RT
003020 P$(6)="(C)":P$(7)="(I)":P$(8)="(N)":P$(9)="(n)":P$(10)="(P)" RT
003025 P$(11)="(O)":P$(12)="(V)":P$(13)="(d)" RT
003030 ITLE5$       ="PART OF SPEECH" JM
003035 ITLE6$       ="DEFINITION"    JM
003060              RT
003110 /* The following two lines are used as formats in the packing */ JM
003160 /* and the unpacking of record fields */ RT
003165              JM
003210 F1$=HEX(A014A001A001A001A0010001A001A0010001A0030009) RT
003260 F2$=HEX(001CA0030009) RT
003310              RT
003360 /* The extract subroutine is called to extract the user's ID */ JM
003410 /* from the system. This ID is used in setting up the protec- */ JM
003460 /* tion bytes in the record currently being modified. */ JM
003510 CALL "EXTRACT" ADDR("ID",USER$, "IL",LIB$, "IV",IVOL$) JM
003560              RT
003610 /* The following code sets up a screen, which asks for the */ JM
003660 /* name of the file containing the common word list that is */ RT000070
003710 /* to be modified. It also provides the opportunity to change */ RT000080
003760 /* the library and volume names which are set to the appro- */ RT000090
003810 /* priate defaults. */ RT000100
003815              RT
003860 FILES="CMNWDLIST" JM
003960 ER1$=" ":E$=HEX(8C): G$,L$,V$=HEX(81) JM
004010 FILMNU: 000500
004060 ACCEPT 000510
004110 AT (01,01), RT000520
004160 "CRES: Common Word List Maintenance Subsystem", RT000530
004210 AT (01,57), RT000540
004260 "(File Selection Screen)", RT000550
004310 AT (02,01), RT000560
004360 "===== RT000570
004410 =====", RT000580
004460 AT (04,14), RT000590
004510 "Welcome to the", 000600
004560 AT (04,29), FAC(HEX(85)), JM
004610 HEADER$, RT000620
004660 AT (07,05), RT
004710 "Please specify the", RT
004760 AT (07,24), FAC(HEX(84)), RT
004810 FIL$, RT
004860 AT (07,41), RT
004910 "you want to modify or create, if the", JM
004960 AT (08,05), RT
005010 "default values are not satisfactory.", RT
005260 AT (13,30), 000770
005310 "File =", RT000780
005360 AT (13,43),FAC(G$),FILES, CH(08), RT000790
005410 RANGE ( ":", " ", "*****"), RT000800
005460 AT (14,30), RT000810
005510 "Library =", RT000820

```

# Technical Report 36-001

Wang VS Integrated Editor - Version 6.10.00 11:35 11/09/84 Page 3  
Input File is CRESQOWL in Librar. CRESQSRC on Volume CRES02

```

005560      AT (14,43),FAC(L$), LIB$          , CH(08),      : RT000830
005610      RANGE ( ":", " ", "*****"),      : RT000840
005660      AT (15,30),      : RT000850
005710 "Volume" =",      : RT000860
005760      AT (15,43),FAC(V$), IVOL$          , CH(06),      : RT000870
005810      RANGE ( ":", " ", "*****"),      : RT000880
005860      AT (18,24), FAC(E$), ER1$,      : RT
005862      AT (20,30),"Press",      : RT
005864      AT (20,36),FAC(HEX(84)),      : RT
005866 ENTER$,      : RT
005868      AT(20,42),      : RT
005870 "to continue",      : RT
005872      AT (21,17),"Press",      : RT
005874      AT (21,23), FAC(HEX(84)),      : RT
005876 PF16$,      : RT
005878      AT (21,28),      : RT
005880 "to return to the Common Word List Menu",      : RT
005890 KEYS(HEX(001020)),      : JM
006110 ON (BIN(16)&BIN(32)) GOTO OUT,OUT      : RT
006210      ,      : RT000960
006260 /* SET UP THE USER FILE BLOCK */      : 000970
006310      : RT000980
006360 SELECT #1,"COMMON",VARC,INDEXED,RECSIZE=40,KEYPOS=1,KEYLEN=20, : RT000990
006410      IOERR GOTO IOERR1      : RT001000
006460      : RT001010
006510 /* SEE IF THE SPECIFIED FILE EXISTS      */ : RT
006560 /* Y- the file exists and the program may continue      */ : JM
006610 /* N- the file doesn't currently exist, you may create one      */ : RT
006660 /* V- the volume supplied doesn't exist or is not mounted      */ : RT
006710 /* L- the library supplied doesn't exist in the volume given      */ : RT
006760 /* E- there is an I/O error in the file specified      */ : RT
006810      : RT001030
006860 CALL "LIMITS"(FILE$,LIB$,IVOL$,CODE$,ER1$)      : RT
006910 IF CODE$="Y" THEN RECCK      : RT
006960 IF CODE$="N" THEN CREATE      : RT
007010 IF CODE$="V" THEN GOSUB'7(STR(ER1$,3),HEX(81),HEX(81),HEX(91))      : RT
007060 IF CODE$="L" THEN GOSUB'7(STR(ER1$,3),HEX(81),HEX(91),HEX(81))      : RT
007110 IF CODE$="E" THEN GOSUB'7(STR(ER1$,3),HEX(81),HEX(81),HEX(81))      : RT
007160 GOTO FILMNU      : 001100
007210      : RT001110
007260 /* The following screen allows the user to create an addition- */ : JM
007310 /* al Common Word List, if the user so desires, OR return to */ : JM
007360 /* the previous screen to respecify necessary file information.*/ : JM
007410      : RT
007460 CREATE:      : RT
007510 ER1$=" "      : RT
007560 ACCEPT      : RT
007610 "CRES: Common Word List Maintenance Subsystem",      : RT
007660      AT (01,53),      : RT
007710 "(Create File Option Screen)",      : RT
007760      AT (02,01),      : RT
007810 "=====":      : RT
007860 "=====":      : RT
007910      AT (06,08),      : RT
007960 "The file you have selected does not currently exist in the libra: RT
008010 ry",      : RT
008060      AT (07,08),      : RT

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 11:35 11/09/84 Page 4  
Input File is CRES0CWL in Library CRES0SRC on Volume CRES02

```

008110 "and volume specified. For your convenience, the following are ", JM
008160 AT (08,08), RT
008210 "the parameters you submitted.", RT
008260 AT (11,30), RT
008310 "File =", RT
008360 AT (11,40),FAC(HEX(85)),FILE$ , CH(08), RT
008410 AT (12,30), RT
008460 "Library=", RT
008510 AT (12,40), FAC(HEX(85)), LIB$ , CH(08), RT
008560 AT (13,30), RT
008610 "Volume =", RT
008660 AT (13,40), FAC(HEX(85)), IVOL$ , CH(06), RT
008710 AT (17,15), RT
008760 "To create a new file from these parameters, press", JM
008810 AT (17,65), FAC(HEX(85)), ENTER$ , CH(05), RT
008860 AT (19,26), RT
008910 "To select another file, press", JM
008960 AT (19,56), FAC(HEX(85)), PF1$ , CH(03), RT
009010 KEY$=BIN(0)&BIN(1)&BIN(17),KEY(OPTION), RT
009060 OR &BIN(1)&BIN(17)) GOTO BLANKF,BLANKF JM
009110 RT
009160 /* Create the new file and insert a header record to identify it*/ RT
009210 OPEN NOGETPARM #1, OUTPUT, FILE= FILE$, LIBRARY=LIB$, RT
009260 VOLUME= IVOL$, SPACE=1000 RT
009310 IOBUFF$= " COMMON WORDS" RT
009360 WRITE #1, IOBUFF$ RT
009410 CLOSE #1 RT
009460 GOTO Z11 JM
009510 RT
009560 BLANKF: FILE$= "GOTO FILMNU RT
009610 /* Check the record size before opening the file to avoid */ RT
009660 /* accidentally resetting it. */ RT
009710 RECC: CALL "READFOR" ADDR(FILE$,LIB$,IVOL$,0%,"RS",RSIZE,REF%) RT
009760 IF RSIZE<>40 THEN WRGSIZE RT
009810 Z11: OPEN NOGETPARM #1,SHARED, FILE= FILE$, LIBRARY = LIB$, : JM
009860 VOLUME = IVOL$ 001150
010160 RT
010210 READ #1, IOBUFF$ RT
010260 IF IOBUFF$=" COMMON WORDS" THEN YES RT
010310 CLOSE #1 JM
010360 WRGSIZE: ES=HEX(94) RT
010410 ER1$= "File type is wrong - please respecify." RT
010460 GOTO FILMNU RT
010510 YES: RT
010560 /* This section of code generates another screen, which asks */ JM
010610 /* for the particular word and its corresponding operation - */ JM
010660 /* either insert word, delete word, or modify word. The user is */ RT001190
010710 /* also at this point given the option of returning to the */ JM
010760 /* Common Word List Menu. */ JM
010810 RT001220
010860 ER$,ER1$,WORD$ = " " RT
010910 ZAP$="Please select the word:"
010960 ES=HEX(84) RT
010961 X$=HEX(81) JM
010965 PASSWRD$=HEX(000102030E101112131E20) JM
011060 CWMENU: 001340
011110 ACCEPT : 001360

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 11:35 11/09/84 Page 5  
Input File is CRES@CWL in Library CRES@SRC on Volume CRES02

```

011160      AT (01,01), "CRES: Common Word List Maintenance Subsystem", RT001370
011210      AT (01,48), "(Word & Option Selection Screen)", RT001380
011260      AT (02,01), "===== RT001390
011310      =====", RT001400
011360      AT (04,08), FAC(HEX(8C)), RT
011410      JAP$, RT001420
011460      AT (06,28), FAC(HEX(8C)), RT
011510      JORDEQS$, RT001440
011560      AT (06,36), FAC(X$), WORDS, CH(20), JM
011610      AT (08,16), FAC(E$), ER$, RT
011660      AT (09,28), FAC(E$), ERIS$, RT
011710      AT (11,08), "To cont: RT
011760      indd, select the PF key corresponding to the desired operation.", RT
011810      AT (13,30), FAC(HEX(A5)), RT001490
011860      PF$, RT
011910      AT (13,38), FAC(HEX(AC)), RT001510
011960      OPER$, RT001520
012010      AT (15,31), FAC(HEX(84)), RT001530
012060      OS$, RT001540
012110      AT (15,39), RT001550
012160      "INSERT word", RT001560
012210      AT (16,31), FAC(HEX(84)), RT001570
012260      IS$, RT001580
012310      AT (16,39), RT001590
012360      "DELETE word", RT001600
012410      AT (17,31), FAC(HEX(84)), RT001610
012460      DS$, RT001620
012510      AT (17,39), RT001630
012560      "MODIFY word", RT001640
012610      AT (18,31), FAC(HEX(84)), JM
012660      JS$, JM
012710      AT (19,39), JM
012760      "Protection Override", RT
012810      AT (21,31), FAC(HEX(84)), RT
012860      JS$, RT001700
012910      AT (21,39), RT
012960      "Return to Common Word List Menu", RT001720
013010      REED(PARM(05)), JM
013060      REED(PARM(06)), JM
013110      IF (OPTION=16 OR OPTION=32) THEN EXIT JM
013160      FOR I = 1 TO DEL(WORD$)
013210      IF (LEN("ABCDEFGHIJKLMNOPQRSTUVWXYZ")=STR(WORD$,I,1))=0 ! JM
013260      THEN ILLERR
013310      NEXT I
013360
013410      ES=HEX(80) RT
013460      ER$,ERIS$="" RT
013510      /* Check for confirmation of deletion */ RT
013560      IF (OPTION=0)AND(D=1) THEN DEL RT
013610      /* Check for cancellation of deletion */ JM
013660      IF D=1 THEN UPDATE RT
013710      /* Check for override of the protection byte */ RT
013760      IF (OPTION=14) OR (OPTION=30) THEN OVERRIDE JM
013810      IF STR(WORD$,1,1)<>" " THEN REED RT
013860      RT001830
013910      RT001840

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 11:35 11/09/84 Page 6  
Input File is CRES@CWL in Library CRES@SRC on Volume CRES02

```

014260 ER$="          The first character cannot be blank."
014310 ER1$= "Please reenter the appropriate word and PF key."      RT001850
014360 E$= HEX(94)                                                RT
014410 GOTO CWMENU                                                RT
014460                                                            001880
014510 ILLERR:
014560 ER$="Illegal character found in word - please reenter."
014610 E$= HEX(94): GOTO CWMENU                                    RT
014660
014710 OVERRIDE:                                                  RT
014760 IF NF=1 AND P#1 THEN CHK                                    RT
014810 ER$="Please check first if the protection is set by selecting PF3" RT
014860 E$=HEX(84)                                                  RT
014910 GOTO CWMENU                                                RT
014960                                                            RT
015010 /* Protection override is confirmed if the protection bytes*/ JM
015060 /* indicate a user not currently logged on or if the        JM
015110 /* current user's id equals the protection bytes.            JM
015160 CHK:
015210 IF (USER$=PROTECT$) OR (WRKST#=-1) THEN REMOVE              RT
015260 ER$= "You don't have the clearance to override the protect." RT
015310 E$= HEX(84)                                                 RT
015360 GOTO CWMENU                                                RT
015410                                                            RT
015460 /* Reset the protection bytes to blanks */                  JM
015510 REMOVE: PROTECT$= " " :P#=0                                  RT
015560 READ #1, HOLD, KEY=WORD$,IOBUFF$                             RT
015610 $PACK (F=F1$) IOBUFF$ FROM WORD$, PRTOFSPC$, PPARIFL$, : RT
015660 NEGF1$,IOBEFL$,ROOTFL$,MWLFL$,PROTECT$                      RT
015710 REWRITE #1, IOBUFF$, DATA GOTO IOERR2                      RT
015760 ER$= "          You may now select your option."           RT
015810 E$= HEX(84)                                                 RT
015860 GOTO CWMENU                                                RT
015910                                                            RT
015960 IOERR1: PRINT                                              001890
016010 E$=HEX(94)                                                  RT
016060 ER1$= "File type is wrong - please respecify."             RT
016110 CLOSE #1                                                    001910
016160 GOTO FILMNU                                                001930
016210                                                            RT001940
016260 IOERR2: PRINT                                              RT001950
016310 PRINT HEX(94),"Error occurred on file "; FILE$             RT001960
016360 PRINT HEX(94),"Error code = "; FS(#1)                     RT001970
016410 STOP "Touch ENTER to return to menu."                     RT001980
016460 GOTO CWMENU                                                RT001990
016510                                                            RT002000
016560 /* IF "ENTER" was hit accidentally before an option was given */ RT
016610 REED: IF OPTION<0> THEN SRCH                                  RT
016660 ER$= "          Command missing."                            RT
016710 E$= HEX(94)                                                 RT
016760 GOTO CWMENU                                                RT
016765                                                            RT
016770 SRCH: P#=1                                                  RT
016810 READ #1,HOLD,KEY=WORD$,IOBUFF$,BOD GOTO NOTFOUND,          : RT
016860 DATA GOTO IOERR2                                           RT002020
016910                                                            RT002030
016960 NF=1                                                        RT002040

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 11:35 11/09/84 Page 7  
Input File is CRESEOWL in Library CRESESRC on Volume CRES02

```

017010                                     RT002660
017060      /* Unpack and check if currently being modified by another */ RT002670
017110                                     RT002680
017160      $UNPACK (F=F1$) IOBUFF$ TO WORD$, PRTOFSPC$, PPARTFL$,      ! RT002690
017210      NEGFL$, TOBEFL$, ROOTFL$, MWLFL$, PROTECT$
017260      IF PROTECT$= " " THEN NOW
017310      CALL "FINDUSER" ADDR( PROTECT$, USERNAME$, WRKST$)
017360      ES= HEX(94)
017410      CONVERT WRKST$ TO WRKST$, PIC(###)
017460      ER$= "      Sorry, "& USERNAME$&" is currently editing "
017510      ER1$= "''"& WORD$ & "'& " at workstation "& WRKST$
017560      REWRITE #1, IOBUFF$
017610      GOTO CWMENU
017660                                     RT
017710 NOW:      PROTECT$= USER$
017760      $PACK (F=F1$) IOBUFF$ FROM WORD$, PRTOFSPC$, PPARTFL$,      ! RT
017810      NEGFL$, TOBEFL$, ROOTFL$, MWLFL$, PROTECT$
017860      REWRITE #1, IOBUFF$, DATA GOTO IOERR2
017910                                     RT
017960      /* NOW BRANCH TO THE OPTION SELECTED */ RT002050
018010 OPT: ON OPTION GOTO INSERT, DELET, MODIFY
018060      ON OPTION-16 GOTO INSERT, DELET, MODIFY
018160                                     RT
018210 NOTFOUND: NF=0
018260      GOTO OPT
018310                                     RT002160
018360 INSERT:                                     RT002170
018410      /* NF=0 indicates the word doesn't yet exist so go ahead */ RT002180
018460      /* and insert. NF=1 indicates the word exists. */ RT002190
018510                                     RT002200
018560      IF NF=1 THEN EXIST
018610      MOD$= "(Insert Screen)"
018615      MESING$= "inserting word"
018660      ROOTFL$, PPARTFL$, PRTOFSPC$, NEGFL$, MWLFL$= " "
018710      PROTECT$= USER$
018760      $PACK (F=F1$) IOBUFF$ FROM WORD$, PRTOFSPC$, PPARTFL$,      ! RT
018810      NEGFL$, TOBEFL$, ROOTFL$, MWLFL$, PROTECT$
018860      WRITE #1, IOBUFF$, DATA GOTO IOERR2
018910      GOTO INS /* goto the screen with the word attributes */ RT
018960                                     RT002210
019010 /* Coming from insert with the word currently existing, */ RT
019060 /* print an error message and erase the protection from */ RT
019110 /* the record. */ RT
019160 EXIST:                                     RT002220
019210      ER$= "The word '" & WORD$ & "' already exists in this list." RT002230
019260      ES= HEX(94)
019310      READ #1, HOLD, KEY= WORD$, IOBUFF$
019360      PROTECT$= " ": P$=0
019410      $PACK (F=F1$) IOBUFF$ FROM WORD$, PRTOFSPC$, PPARTFL$,      ! RT
019460      NEGFL$, TOBEFL$, ROOTFL$, MWLFL$, PROTECT$
019510      REWRITE #1, IOBUFF$, DATA GOTO IOERR2
019560      GOTO CWMENU
019610                                     RT
019660 /* DELET, DEL, & NODEL all pertain to the delete option */ RT
019710 /* DELET- gives the user a chance to cancel the deletion */ JM
019760 DELET: IF NF=0 THEN NODEL
019810      DF=1
019810                                     RT002420
019810                                     RT002430

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 11:35 11/09/84 Page 8  
Input File is CRES@CWL in Library CRES@SRC on Volume CRES02

```

019860      ER$= "          Press ENTER to confirm deletion of "& WORD$ RT
019910      ER1$="or press PF14 to cancel the deletion. " JM
019960      ES= HEX(94) RT
019965      PASSWRD$=HEX(000E1E) JM
019970      XS=HEX(85) JM
020010      GOTO CWMENU RT002460
020060 RT002470
020110 /* DEL- deletes the entire record */ RT
020160 DEL: READ #1, HOLD,KEY=WORD$,IOBUFF$ RT
020210 DELETE #1 RT
020260 DF=0 RT002490
020310 ER$= " '&WORD$&' deleted - Please select your next word." RT
020360 ZAP$=" " JM
020410 WORD$=" " :NF=0 JM
020420 PASSWRD$=HEX(000102030E101112131E20) JM
020425 XS=HEX(81) JM
020460 GOTO CWMENU RT002530
020510 002540
020560 /* NODEL- prints an error messages when the word doesn't exist */ RT
020610 NODEL: ER$= "The word '& WORD$ &' does not exist in this list." RT002550
020660 ES= HEX(94) RT
020710 GOTO CWMENU RT002570
020760 RT002580
020810 MODIFY: IF NF=1 THEN GETIT RT002590
020860 ER$="The word you are attempting to modify does not exist"& : RT
020910 " in this list." RT001300
020960 /* if word not in list, */ RT
021010 ES=HEX(94) /* set up message and fac and */ RT
021060 GOTO CWMENU /* go back up to previous screen */ RT002620
021110 RT002630
021160 GETIT: RT
021210 RT002720
021260 MOD$="(Modify Screen)" RT002730
021270 MESING$="modifying word", JM
021310 INS: RT002740
021360 ER1$=" " RT
021410 ES=HEX(8C) RT
021460 RT002830
021510 /* This accept statement generates a screen with the attributes */ RT
021560 /* corresponding to each word appropriately labeled for insert */ RT
021610 /* or modification. It also calls another screen on request if */ JM
021660 /* there is a need for a explanation of legal entries for the */ RT
021710 /* parts of speech. */ RT
021760 AOK: RT002840
021810 ACCEPT : RT002850
021860 AT (01,01), : RT002860
021910 "CRES: Common Word List Maintenance Subsystem", : RT002870
021960 AT (01,66), FAC(HEX(8C)),MOD$, : RT002880
022010 AT (02,01), "===== RT002890
022060 =====", RT002900
022110 AT (03,07), : RT002910
022160 "For an explanation of legal entries, press", : JM
022210 AT (03,50),FAC(HEX(84)), PF1$, : RT
022260 AT (03,54), : RT
022310 "; otherwise, enter", : RT002950
022360 AT (04,07), : RT002960
022410 "the appropriate attribute(s).", : RT002970

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 11:35 11/09/84 Page 9  
Input File is CRES0CWL in Library CRES0SRC on Volume CRES02

```

022460      AT (06,33),FAC(HEX(AD)), DEF$ ,      : RT002980
022510      AT (06,41),FAC(HEX(AD)), A5$ ,      : RT002990
022560      AT (06,58),FAC(HEX(AD)), LEG$ ,      : RT003000
022610      AT (06,64),FAC(HEX(AD)), A7$ ,      : RT003010
022660      AT (08,07),      : RT003020
022710 "WORD:",      : RT003030
022760      AT (08,15),FAC(HEX(84)), WORD$,      : RT003040
022810      RANGE ( ":", " ", "cccccccccccc"),      : RT003050
022860      AT (08,61),      : RT003060
022910 "(A-Z.-')",      : JM
022960      AT (10,07),      : RT003080
023010 "PART OF SPEECH:",      : RT003090
023060      AT (10,41),FAC(HEX(80)),PRTOFSPC$, CH(01),      : RT003100
023110      AT (10,57),      : RT003110
023160 "(ABCDEFGHIJNOPVadjn)",      : RT
023210      AT (11,07),      : RT003130
023260 "PAST PARTICIPLE:",      : RT003140
023310      AT (11,41), PPARTFL$ , CH(01),      : RT003150
023360      AT (11,57),      : RT003160
023410 "(* or NO ENTRY)",      : RT003170
023460      AT (12,07),      : RT003180
023510 "NEGATIVE:",      : RT003190
023560      AT (12,41), NEGFL$ , CH(01),      : RT
023610      AT (12,57),      : RT003210
023660 "(* or NO ENTRY)",      : RT003220
023710      AT (13,07),      :
023760 "TO BE:",      :
023810      AT (13,41), TOBEFL$ , CH(01),      :
023860      AT (13,57),      :
023910 "(* or NO ENTRY)",      :
023960      AT (14,07),      :
024260 "ROOT WORD:",      :
024310      AT (14,41), ROOTFL$, CH(01),      : RT
024360      AT (14,57),      : RT
024410 "(* or NO ENTRY)",      :
024460      AT (17,25),FAC(E$),ER1$,      : RT
024510      AT (20,17),"Use",FAC(HEX(85)),TAB$,"or",FAC(HEX(85)),      : RT
024560      BTAB$,"to move between modifiable fields.",      : JM
024610      AT (23,30),      : RT
024660 "Press",      : RT003270
024710      AT (23,36),FAC(HEX(84)),ENTER$,      : RT
024760      AT (23,42),      : RT
024810 "to continue",      : RT003300
024860      AT (24,21),      : JM
024910 "Press", AT(24,27),FAC(HEX(84)),PF16$,AT(24,32),      : JM
024960 "to exit without ",      : JM
024965 AT(24,48),FAC(HEX(8C)), MESING$,      : JM
025010 KEYS(BIN(0)&BIN(1)&BIN(17)&BIN(16)&BIN(32)),KEY(OPTION),      : RT
025060 ON (BIN(16)&BIN(32)) GOTO EXIOM,EXIOM      : RT
025110      : RT003320
025310 IF OPTION=1 OR OPTION=17 THEN DEFIN      : RT
025360 ER1$=" "      : RT
025410      : 003380
025460 /* Check for validity of entries */      : RT003390
025510      : RT003400
025560      IF POS("ABCDEFGHIJNOPVadjn"=PRTOFSPC$)>0 THEN NEXT1      : RT
025610      ER1$ = "Part of Speech value is illegal."      : RT003420

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 11:35 11/09/84 Page 10  
Input File is CRES0CWL in Library CRES0SRC on Volume CRES02

```

025660      ES = HEX(94)                                RT
025710      GOTO AOK                                    RT003440
025760
025810 NEXT1:  IF POS(" **=PPARTFL$)>0 THEN NEXT2      RT
025860      ER1$ = "Past Participle value is illegal."  RT003450
025910      ES = HEX(94)                                RT003460
025960      GOTO AOK                                    RT
026010 NEXT2:  IF POS(" **=NEGFL$)>0 THEN NEXT3        RT003480
026060      ER1$ = " Negative value is illegal."        RT
026110      ES = HEX(94)                                RT
026160      GOTO AOK                                    RT003520
026210 NEXT3:  IF POS(" **=TOBEFL$)>0 THEN NEXT4        RT
026260      ER1$ = " To Be value is illegal."           JM
026310      ES = HEX(94)                                RT
026360      GOTO AOK                                    RT
026610 NEXT4:  IF POS(" **=ROOTFL$)>0 THEN UPDATE      RT
026660      ER1$ = " Root word value is illegal."      RT
026710      ES = HEX(94)                                RT
026760      GOTO AOK                                    RT
026810      /
026860 /* EXIOM IS USED TO EXIT THE INSERT MODE AND CALL NOCHAN */ RT
026910 /* TO EXIT THE MODIFY MODE */                  RT
026960 EXIOM:  READ #1, HOLD, KEY=WORD$, IOBUFF$      RT
027010      IF NF=1 THEN NOCHAN                        RT
027060      DELETE #1                                    RT
027110      ER$ = " Word "&WORD$&" was not inserted."  RT
027160      GOTO CWMENU                                RT
027210
027260 NOCHAN: PROTECT$ = " " :P%=0                  RT003530
027310      $PACK (F=F2$) IOBUFF$ FROM PROTECT$      RT
027360      REWRITE #1, IOBUFF$, DATA GOTO IOERR2      RT
027410      ER$ = " No changes were saved."            RT
027460      ES= HEX(8C)                                RT
027510      GOTO CWMENU                                RT
027560
027610 /* UPDATE removes the protection on the record and writes out */ RT
027660 /* the final data as input. */                  JM
027710 UPDATE:  READ #1, HOLD, KEY=WORD$, IOBUFF$      RT
027760      PROTECT$ = " " :P%=0                        RT
027810      $PACK (F=F1$) IOBUFF$ FROM WORD$, PRTOPSPC$, PPARTFL$, ! RT
027860      NEGFL$, TOBEFL$, ROOTFL$, MWLFL$, PROTECT$  RT
027910      REWRITE #1, IOBUFF$, DATA GOTO IOERR2      RT003600
027960      ER1$ = " "
028010      IF NF=0 THEN RITE /* NF=0 indicates insert mode */ RT
028060      IF DF=1 THEN EXDEL /* DF=1 indicates delete mode */ RT
028110      ZAP$ = " " /* otherwise in modify mode */ RT
028160 ER$ = " "&WORD$&" updated - Please select next word." JM
028210      ES=HEX(8C)                                    RT
028260      WORD$ = " "
028310      GOTO CWMENU                                RT003610
028360 RITE:
028410 ER$ = " "&WORD$&" inserted - Please select your next word." RT003620
028460      ZAP$ = " "
028510      WORD$ = " "
028560      GOTO CWMENU                                RT003660
028610
028660 EXDEL:  DF=0

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 11:35 11/09/84 Page 11  
Input File is CRES@CWL in Library CRES@SRC on Volume CRES02

```

028710      ER$="          Deletion cancelled."          JM
028720      PASSWRD$=HEX(000102030E101112131E20)          JM
028725      X$=HEX(81)          JM
028760      GOTO CWMENU          RT
028810          RT
029110          003720
029160 /* THIS FUNCTION PRINTS OUT ERROR MESSAGES AND */          003730
029210 /* SETS UP FLASHING FAC'S IN APPROPRIATE PLACES */          003740
029260 DEF FN'7(ER1$,B$,C$,D$)          RT
029310      PRINT          003760
029360      E$=HEX(94)          RT
029410      G$=B$          003780
029460      L$=C$          003790
029510      V$=D$          003800
029560 RETURN          RT003810
029610          RT003820
029660 DEFIN:          RT003830
029710 ACCEPT          : RT003840
029760      AT (01,01),          : RT003850
029810 "CRES: Common Word List Maintenance Subsystem",          : RT003860
029860      AT (01,51), FAC(HEX(8C)),          : JM
029910 EXP$,          : RT003880
029960      AT (02,01), "=====          RT003890
030010 =====",          : RT003900
030055      AT (05,08), FAC(HEX(AD)),      ITLE5$      , CH(14), : JM
030060      AT (05,49), FAC(HEX(AD)),      ITLE6$      , CH(10), : JM
030065      AT (07,03), FAC(HEX(84)),      P$(1)      , CH(03), : RT
030070      AT (07,08),          : RT
030075 "Adjective .....Modifies a noun or pronoun",          : RT
030080      AT (08,03), FAC(HEX(84)),      P$(2)      , CH(03), : RT
030085      AT (08,08),          : RT
030090 "Adjective/verb .....A word that may function as an adjective          RT
030095 or verb",          : RT
030100      AT (09,03), FAC(HEX(84)),      P$(3)      , CH(03), : RT
030105      AT (09,08),          : RT
030110 "Adverb .....Modifies a verb, adjective, or adverb",          : RT
030115      AT (10,03), FAC(HEX(84)),      P$(4)      , CH(03), : RT
030120      AT (10,08),          : RT
030125 "Adverb/verb .....A word that may function as an adverb or          RT
030130 verb",          : RT
030135      AT (11,03), FAC(HEX(84)),      P$(5)      , CH(03), : RT
030140      AT (11,08),          : RT
030145 "Article .....A limiting adjective before a noun (e.g.          RT
030150 a, the)",          : RT
030155      AT (12,03), FAC(HEX(84)),      P$(6)      , CH(03), : RT
030160      AT (12,08),          : RT
030165 "Conjunction .....Links phrases or clauses (e.g. or)",          : RT
030170      AT (13,03), FAC(HEX(84)),      P$(7)      , CH(03), : RT
030175      AT (13,08),          : RT
030180 "Interjection .....Used for simple exclamations (e.g. Oh!)          RT
030185 ,          : RT
030190      AT (14,03), FAC(HEX(84)),      P$(8)      , CH(03), : RT
030195      AT (14,08),          : RT
030200 "Noun .....Indicates a person, place, or thing",          JM
030205      AT (15,03), FAC(HEX(84)),      P$(13)     , CH(03), : RT
030210      AT (15,08),          : RT
030215 "Noun/adjective .....A word that may function as a noun or ad          RT

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 11:35 11/09/84 Page 12  
Input File is CRES@CWL in Library CRES@SRC on Volume CRES02

```

030220 jective", : RT
030225 AT (16,03), FAC(HEX(84)), P$(9) , CH(03), : RT
030230 AT (16,08), : RT
030235 "Noun/verb .....Word that may function as a noun or verb: RT
030240 ", : RT
030245 AT (17,03), FAC(HEX(84)), P$(10) , CH(03), : RT
030250 AT (17,08), : RT
030255 "Preposition .....Expresses a relationship between two ele: JM
030260 ments of ", : JM
030261 AT (18,08), : JM
030262 " a sentence (e.g. in, on, by, to, since) : JM
030263 ", : JM
030265 AT (19,03), FAC(HEX(84)), P$(11) , CH(03), : JM
030270 AT (19,08), : JM
030275 "Pronoun .....Word that may be used as a substitute fo: RT
030280 r a noun", : RT
030285 AT (20,08), : JM
030290 " (e.g. I, you, she, what,....)", : JM
030295 AT (21,03), FAC(HEX(84)), P$(12) , CH(03), : JM
030300 AT (21,08), : JM
030305 "Verb .....Denotes an action or occurrence", : RT
030315 AT (23,20), : RT
030320 "Press", : RT
030325 AT (23,26), FAC(HEX(84)), PF1$ , CH(03), : RT
030330 AT (23,30), : RT
030335 "to return to the previous screen", : RT
031560 KEYS(BIN(1)&BIN(17)), ON (BIN(1)&BIN(17)) GOTO AOK,AOK RT004190
031610 : RT004200
031660 EXIT: CLOSE #1 004210
031710 OUT: /* RETURNING TO COMMON WORD LIST MENU */ JM

```

Print Common Word List (CRES@CWP)

CRES@CWP in library CRES@SRC is the BASIC language program to print the Common Word List. The object language version is CRES@CWP in library CRES@SAV.

This program first allows the user to specify the word list to be printed. Then the program enables the user to print out either the entire Common Word List, or a specified portion. This printout can be displayed at the terminal or sent to the printer. An optional subtitle of up to two lines can be included for the printout, and the number of lines per page may be specified as some value less than the default value of 50.

A complete program listing follows. Appendix C contains a sample flowchart.

Technical Report 86-001

THIS PAGE INTENTIONALLY LEFT BLANK.

Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 15:41 11/08/84 Page 1  
Input File is CRES@CWP in Library CRES@SRC on Volume CRES02

```

000010 /* This is the CRES COMMON WORD LIST PRINT SUBSYSTEM PROGRAM */ JM
000020 /* */ RT
000030 /* AUTHORS: Rose Tank and John Mears */ RT
000040 /* Date: October 12, 1983 */ RT
000050 /* */ RT
000060 */ RT
000070 /* The following code sets up a screen, which asks for the */ RT
000080 /* name of the file containing the common word list */ RT
000090 /* to be printed. It also provides the opportunity to change */ RT
000100 /* the library and volume names which are set to the appro- */ RT
000110 /* priate defaults through the "Extract" system routine. */ RT
000120 */ RT
000130 */ RT
000140 DIM WORD$24,ERR1$50,CODE$1,MSG$50,F$40,G$1,CMSG$255,MSG2$40 RT
000150 DIM F1$20,IOWUFF$64,EXP$31,PFILE$08,RLIB$8,RVOL$6 JM
000160 DIM TAB$3,BTAB$7,FILE$08,LIB$08,ILIB$08,PLIB$8,PVOL$6 JM
000170 DIM IVOL$06,HEADER$51,L$1,V$1,B$1,C$2,D$8,M$1,MSK$2 RP
000180 DIM SBTITLE$20,A$(14)10,SBITL$20,E2$44 JM
000190 DIM ENS 05 RT
000200 DIM EX$ 04 RT
000210 DIM ER$ 79,MESSG$21 RT
000220 DIM STARIPH$ 20 RT
000230 DIM ENDPH$ 20 RT
000240 DIM TP$ 08 RT
000250 DIM PF1$ 04 RT
000260 DIM PF16$ 04 RT
000270 RT
000280 ENS ="ENTER" RT
000290 A$(1)= "adjective" RT
000300 A$(2)= "verb" RP
000310 A$(3)= "conjunct'n" RT
000320 A$(4)= "adj/verb" RT
000330 A$(5)= "noun" RP
000340 A$(6)= "pronoun" RT
000350 A$(7)= "interject" RT
000360 A$(8)= "noun/verb" RT
000370 A$(9)= "advero" RT
000380 A$(10)="preposit" RT
000390 A$(11)="article" RT
000400 A$(12)="adv/verb" RT
000410 A$(13)="noun/adj" RT
000420 RT
000430 TAB$="TAB" RT
000440 BTAB$="BACKTAB" RT
000450 EXP$="(Explanation of Entries Screen)" RT
000460 MESSG$="(Parts of Speech Codes)" RT
000470 F1$=HEX(0014000100010001000JA00100010003000B) RT
000480 F$= "Filename not found on (library/volume)." RT
000490 ERR1$= "I/O ERROR - Please respecify a COMMON word list." RT
000495 E2$= " Please specify a COMMON word list." JM
000500 CALL "EXTRACT" ADDR("IV",IVOL$,"IL",ILIB$,"RV",RVOL$,"SL",PLIB$, : JM
000510 "SV",PVOL$,"RL",RLIB$) JM
000520 FILE$="CMNWLST"
000530 LIB$=ILIB$ JM
000540 G$=HEX(81) JM
000550 MSG$=" ":L$=HEX(81) RT
000560 V$=HEX(81): M$=HEX(8D) RT

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 15:41 11/08/84 Page 2  
Input File is CRES@CWP in Library CRES@SRC on Volume CRES02

```

000570 HEADERS$      ="CRES COMMON WORD LIST PRINT SUBSYSTEM"      RT
000580 FIL$          ="Common Word List"                            JM
000590 EX$="PF16"                                             RT
000600                                                         RT
000610 FILMNU:                                             RT
000620 ACCEPT                                             : RT
000630      AT (01,01),                                       : RT
000640 "CRES: Common Word List Print Subsystem",           : RT
000650      AT (01,57),                                       : RT
000660 "(File Selection Screen)",                             : RT
000670      AT (02,01),                                       : RT
000680 "=====RT
000690 =====",                                             RT
000700      AT (04,15),                                       : JEM
000710 "Welcome to the",                                       : RT
000720      AT (04,26), FAC(HEX(85)),                           : RT
000730 HEADERS$,                                             RT
000740      AT (07,12),                                       : JM
000750 "Please specify the",                                       : JM
000760      AT (07,31), FAC(HEX(84)),                           : JM
000770 FIL$,                                             : JEM
000780      AT (07,48),                                       : JM
000790 "you want to print,",                                       : JM
000800      AT (08,18),                                       : JM
000810 "if the default values are not satisfactory.      ",      RT
000820      AT (13,30),                                       : RT
000830 "File      =",                                       : RT
000840      AT (13,43),FAC(G$),FILES$      , CH(08),           : RT
000850      RANGE ( "!"      , "*****"),                   : RT
000860      AT (14,30),                                       : RT
000870 "Library      =",                                       : RT
000880      AT (14,43),FAC(L$), LIB$      , CH(08),           : RT
000890      RANGE ( "!"      , "*****"),                   : RT
000900      AT (15,30),                                       : RT
000910 "Volume      =",                                       : RT
000920      AT (15,43),FAC(V$), IVOL$      , CH(06),           : RT
000930      RANGE ( "!"      , "*****"),                   : RT
000940      AT (18,24), FAC(M$), MSG$,                           : RT
000950      AT (20,30),"Press",                                   : RT
000960      AT (20,36),FAC(HEX(84)),                           : RT
000970 EN$,                                             : RT
000980      AT(20,42),                                       : RT
000990 "to continue",                                       : JM
001000      AT (21,17),"Press",                                   : RT
001010      AT (21,23), FAC(HEX(84)),                           : RT
001020 EX$,                                             : RT
001030      AT (21,28),                                       : RT
001040 "to return to the Common Word List Menu",           : JM
001050 ON (BIN(16)&BIN(32)) GOTO OUT      RT
001070                                                         RT
001080 /* SET UP THE USER FILE BLOCK */      RT
001090                                                         RT
001100 SELECT #1,"COMMON",VARC,INDEXED,RECSIZE=40,KEYPOS=1,KEYLEN=20, : RT
001110      IOERR GOTO IOERR1      RT
001120                                                         RT
001130 SELECT #2,"PRTFILE",PRINTER,RECSIZE=80,IOERR GOTO IOERR1      RT
001140                                                         RT

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 15:41 11/08/84 Page 3  
Input File is CRES02WP in Library CRES02 on Volume CRES02

```

001150 /* SEE IF THE SPECIFIED FILE EXISTS */ RT
001160 RT
001170 CALL "LIMITS"(FILE$,LIB$,IVOL$,CODE$,MSG$) RT
001180 IF CODE$="Y" THEN ZZZ RT
001190 IF CODE$="N" THEN GOSUB'7(F$,HEX(91),HEX(81),HEX(81)) RT
001200 IF CODE$="V" THEN GOSUB'7(STR(MSG$,3),HEX(81),HEX(81),HEX(91)) RT
001210 IF CODE$="L" THEN GOSUB'7(STR(MSG$,3),HEX(81),HEX(91),HEX(81)) RT
001220 IF CODE$="E" THEN GOSUB'7(STR(MSG$,3),HEX(81),HEX(81),HEX(81)) RT
001230 GOTO FILMNU RT
001240 RT
001250 /* IF WE FALL HERE, THE FILE EXISTS - BUT IS IT A COMMON LIST? */ JM
001260 JM
001270 ZZZ: CALL "READFOR" ADDR(FILE$,LIB$,IVOL$,0$,"RS",RAG$,REP$) JM
001290 IF RAG$=40 THEN ZZZ JM
001295 GOSUB'7(RZ$,HEX(91),HEX(81),HEX(81)) JM
001297 GOTO FILMNU JM
001300 RT
001310 ZZZ: OPEN NOBETPARM #1,SHARED, FILE=FILE$, LIBRARY = LIB$, : JM
001320 VOLUME = IVOL$ RT
001330 RT
001340 READ #1,IOBUFF$, DATA GOTO IOERR2 RT
001360 IF STR(IOBUFF$,1,16)=" COMMON WORDS " THEN YES JM
001370 CLOSE #1 RT
001380 M$=HEX(94) RT
001390 MSG$=ERRR$ RT
001400 GOTO FILMNU RT
001410 YES: RT
001420 /* This section of code generates the second screen, which asks */ RT
001430 /* for the particular word interval and its corresponding */ RT
001440 /* print operation. The user is also given the option of */ JM
001460 /* returning to the common word list menu. */ JM
001470 RT
001490 ER$=" " RT
001540 CWMENU: RT
001550 WORD$=" " RT
001560 LPP$=50 RT
001570 PF15$="PF15" RT
001580 PF16$="PF16" RT
001590 SBTITLE$=" " RT
001600 SBTITLE1$=" " RT
001610 BALONEY: RT
001620 TP$="TERMINAL" RT
001630 STARTPH$="FIRST" RT
001640 ENDPH$="LAST" RT
001650 PRINTSCR: RT
001660 ACCEPT : RT
001670 AT (01,03), : RT
001680 "CRES: Common Word List Print Subsystem", : RT
001690 AT (01,66), : RT
001700 "(Print Screen)", : RT
001710 AT (02,03), : RT
001720 "=====": RT
001730 =====: RT
001740 AT (04,13), : RT
001750 "To print a range of lines, fill in the information below.", : RT
001760 AT (06,02), PAC(HEX(85)), : RT
001770 ER$, : RT

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 15:41 11/08/84 Page 4  
Input File is CRES0CWP in Library CRES0SRC on Volume CRES02

```

001780      AT (08,22),           : RT
001790 "Subtitle of list =",   : RT
001800      AT (08,42), FAC(HEX(81)),SBTITLE$,CH(20), : RT
001810      AT (09,42), FAC(HEX(81)),SBTITLE$,CH(20), : RT
001820      AT (12,22),           : RT
001830 "Starting word =",       : RT
001840      AT (12,42),FAC(HEX(81)), STARTPH$, CH(20), : RT
001850      RANGE ( ":", " ", "*****"), : RT
001860      AT (13,22),           : RT
001870 "Ending word =",         : RT
001880      AT (13,42), FAC(HEX(81)), ENDPH$, CH(20), : RT
001890      RANGE ( ":", " ", "*****"), : RT
001900      AT (14,22),           : RT
001910 "Lines per page =",       : RT
001920      AT (14,42), FAC(HEX(82)), LPP%, PIC(##), : RT
001930      AT (17,22),           : RT
001940 "Print at (TERMINAL/PRINTER) =", : RT
001950      AT (17,53), FAC(HEX(81)), TP$, CH(0d), : RT
001960      AT (22,32),           : JM
001970 "Press",                 : RT
001980      AT (22,38), FAC(HEX(85)), EN$, CH(05), : JM
001990      AT (22,44),           : JM
002000 "to print",               : RT
002010      AT (23,17),           : JM
002020 "Press",                 : RT
002030      AT (23,23), FAC(HEX(84)), PF16$, CH(04), : JM
002040      AT (23,28),           : JM
002050 "to return to the Common Word List Menu", : JM
002060 KEYS(BIN(0)&BIN(16)&BIN(32)), : JM
002070 ON (BIN(16)&BIN(32)) GOTO EXIT,EXIT : JM
002080      IF STR(TP$,1,1) = "P" THEN NEXT11 : RT
002090      IF STR(TP$,1,1) <> "P" THEN TERMPR : RT
002100 NEXT11:      IF LPP% < 51 THEN NEXT13 : RT
002110      ER$="Lines per page' can be at most 50." : RT
002120      GOTO PRINTSCR         : RT
002160 :                           : RT
002170 IOERR1: PRINT           : RT
002180      PRINT HEX(94),ERR1$ : RT
002190      CLOSE #1             : RT
002200      FOR I=1 TO 200000 :NEXT I : RT
002210      GOTO FILMENU         : RT
002220 :                           : RT
002230 TERMPR: : RT
002240 ER$="Print at ' has only legal values TERMINAL or PRINTER." : RT
002250      GOTO PRINTSCR         : RT
002260 :                           : RT
002270 REX: ER$="      'Starting word' comes after 'Ending word'." : RT
002280      GOTO PRINTSCR         : RT
002290 :                           : RT
002300 :                           : RT
002310 IOERR2: PRINT           : RT
002320      PRINT HEX(94),"Error occurred on file "; FILE$ : RT
002330      PRINT HEX(94),"Error code = "; FS(1) : RT
002340      STOP "Touch ENTER to return to menu." : RT
002350      GOTO CWMENU           : RT
002360 :                           : RT
002370 NEXT13: IF STARTPH$="FIRST" OR ENDPH$="LAST" THEN FORD : RT

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 15:41 11/08/84 Page 5  
Input File is CRES@CWP in Library CRES@SRC on Volume CRES02

```

002380      IF STARTPH$>ENDPH$ THEN REX      RT
002390      RT
002400 /* Now it's time to generate our print file name */      RT
002410      RT
002420 FORD:  DAT$= DATE      RT
002430      D$= STR(DAT$,3,2)&"/"&STR(DAT$,5,2)&"/"&STR(DAT$,1,2)      RT
002440      T$= STR(TIME,1,2)&":"&STR(TIME,3,2)      RT
002450      RT
002460      PFILE$="COMMAND##"      RT
002470      I%=0      RT
002480 NEXTI4: CONVERT I% TO STR(PFILE$,7,2),PIC(00)      RT
002490      I%=I%+1      RT
002500      CALL "LIMITS" (PFILE$,PLIB$,PVOL$,CODE$,MSG$)      JM
002510      IF CODE$="Y" THEN NEXTI4      RT
002512      IF CODE$="E" THEN ER$ = STR(MSG$,3)      JM
002514      IF CODE$="E" THEN PRINTSCR      JM
002520 DISPLAY      :
002530      AT (01,03),      : RT
002540 "CRES: Common Word List Print Subsystem",      : RT
002550      AT (01,61),      : RT
002560 "(Generation Screen)",      : RT
002570      AT (02,03),      : RT
002580 "=====      : RT
002590 =====",      : RT
002600      AT (12,23),      :
002610 "Output file now being generated:"      :
002620 OPEN NOGETPARM #2,OUTPUT,FILE=PFILE$,LIBRARY=PLIB$,      : JM
002630      VOLUME=PVOL$,SPACE=15000      JM
002640 GOSUB TPL PGE      RT
002650 IF STARTPH$ = "FIRST" THEN STARTPH$ = ALL(HEX(00))      RT
002660 IF ENDPH$ = "LAST" THEN ENDPH$ = ALL(HEX(FF))      RT
002670 PAGE% = 1      RT
002680 RECS% = 1      RT
002690 READ #1, KEY>=STARTPH$,IOBUFF$      RT
002700 GOSUB HEADER      RT
002710 IF STR(IOBUFF$,1,20)>ENDPH$ THEN HOME      RT
002720 GOTO KNOT      RT
002730 AGAN:  READ #1, IOBUFF$,HOD GOTO HOME      RT
002740 IF STR(IOBUFF$,1,20)>ENDPH$ THEN HOME      RT
002750 IF RECS% = 1 AND PAGE% <> 1 THEN GOSUB HEADER      JM
002760 KNOT:  IF STR(IOBUFF$,1,20)=" COMMON WORDS" THEN AGAN      RT
002770 PSN= POS("JVCJNOInApBad"= STR(IOBUFF$,21,1))      RT
002780 IF PSN=0 THEN PSN=14      JM
002790 IF PSN=14 THEN AS(PSN)=STR(IOBUFF$,21,1)      JM
002800 IF STR(TP$,1,1)="T" THEN BAG      RT
002810 WRITE #2, USING CAT6,C$, " ",STR(IOBUFF$,1,20),      : RT
002820      AS(PSN),STR(IOBUFF$,22,1),STR(IOBUFF$,24,1),      : RT
002830      STR(IOBUFF$,23,1)      RT
002840 IF(STR(IOBUFF$,26,1)<>"") THEN BAG      RT
002850 C$=HEX(0000)      RT
002860 WRITE #2, USING CAT9, C$,STR(IOBUFF$,1,20)      RT
002870 WRITE #2, USING CAT9, C$,STR(IOBUFF$,1,20)      RT
002880 C$=HEX(0001)      RT
002890 GOTO BAG      RT
002900 BAK:  IF STR(IOBUFF$,26,1)="*" THEN STR(IOBUFF$,26,1)="+"      RT
002910 WRITE #2, USING CAT6,C$,STR(IOBUFF$,26,1),STR(IOBUFF$,1,20),      : RT
002920      AS(PSN),STR(IOBUFF$,22,1),STR(IOBUFF$,24,1),      : RT

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 15:41 11/08/84 Page 6  
Input File is CRES@CWP in Library CRES@SRC on Volume CRES02

```

002930          STR(IOBUFF$,23,1)                                RT
002940 BAG:      RECS%=RECS%+1                                    RT
002950          IF RECS%>LPP% THEN RECS%=1                        RT
002960          IF RECS%=1 THEN PAGE%=PAGE%+1                    RT
002970          GOTO AGAN                                          RT
002980 HOME:     IF RECS%=1 THEN RECS%=51                        RT
002990          WRITE #2, USING DOG, C$                          RT
003000          RECS%=RECS%+2                                      RT
003010          WRITE #2, USING DG, C$                           RT
003020          FOR I = RECS% TO LPP%+1                          RT
003030          WRITE #2, USING DOG, C$
003040          NEXT I
003050          IF STR(TP$,1,1)="P" THEN HER1
003060          WRITE #2, USING CAT8, C$
003070          GOTO HER
003080 HER1:     WRITE #2, USING RAT, C$
003090          C$=HEX(0000)
003100          FOR I=1 TO 2                                      RT
003110          WRITE #2, USING RAT1, C$
003120          NEXT I
003130          C$=HEX(0001)
003140 HER:     CLOSE #2
003150          IF STR(TP$,1,1) = "P" THEN MESS                  RT
003160          CLOSE WS
003170          MSG2$ = "CANCEL Processing"
003180          MSK$=HEX(00)
003190          CMSG$ = " "
003200          CALL "LINK" ADDR("DISPFILE","P",RLIB$,RVOL$,3%,PFILE$, : JM
003210          PLIB$,      PVOL$,"C",MSG2$,27%,"H",MSK$,CMSG$,255%,C%,R%) JM
003220          ER$ = " "
003230          IF C%<>0 OR R%<>0                                    !
003240          THEN ER$ = STR(CMSG$,141)
003250          GOTO CWMENU
003260
003270 MESS: ER$="      File "&PFILE$&" in Library "& PLIB$ & " on " : JM
003280      &PVOL$ &" generated."
003290          GOTO BAIONEY
003300
003370 TTL PGE:
003380          C$=HEX(8001)
003390          WRITE #2, USING DOG, C$
003400          C$ = HEX(0015)
003410          WRITE #2, USING DOG, C$
003420          C$ = HEX(0001)
003430          WRITE #2, USING DOG1, C$
003440          WRITE #2, USING DOG2, C$
003450          WRITE #2, USING DOG0, C$,D$,I$
003460          WRITE #2, USING DOG3, C$
003470          WRITE #2, USING DOG2, C$
003480          WRITE #2, USING DOG1, C$
003490          WRITE #2, USING DOG2, C$
003500          IF SBTITLE$ = "      " THEN GERRY
003510          WRITE #2, USING DOG4, C$, SBTITLE$
003520          IF SBTITLE1$<>"      " THEN WRITE #2, USING : RT
003530 DOG10, C$, SBTITLE1$
003540          WRITE #2, USING DOG2, C$
003550 GERRY: WRITE #2, USING DOG5, C$, FILE$

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 15:41 11/08/84 Page 7  
Input File is CRES@CWP in Library CRES@SRC on Volume CRES02

```

003560      WRITE #2, USING DOG2, C$                      RT
003570      WRITE #2, USING DOG6, C$, LIB$                RT
003580      WRITE #2, USING DOG2, C$                      RT
003590      WRITE #2, USING DOG7, C$, IVOL$              RT
003600      WRITE #2, USING DOG2, C$                      RT
003610      WRITE #2, USING DOG8, C$, STARTPH$           RT
003620      WRITE #2, USING DOG2, C$                      RT
003630      WRITE #2, USING DOG9, C$, ENDPH$            RT
003640      WRITE #2, USING DOG2, C$                      RT
003650      WRITE #2, USING DOG1, C$                     RT
003660      RETURN                                          RT
003670                                          RT
003680  HEADER:                                          RT
003690      IF PAGE%=1 THEN FIRST
003700      C$ = HEX(0001)
003710      IF STR(TP$,1,1)="P" THEN FRST1
003720      WRITE #2, USING DOG, C$
003730      WRITE #2, USING CAT8, C$
003740      GOTO FIRST
003750  FRST1: WRITE #2, USING RAT, C$
003760      C$=HEX(0000)
003770      FOR I=1 TO 2
003780      WRITE #2, USING RAT1, C$                        RT
003790      NEXT I
003800      C$=HEX(0001)
003810  FIRST: C$ = HEX(8001)
003820      WRITE #2, USING DOG, C$                        RT
003830      C$ = HEX(0001)
003840      IF SBTITLE$ = "
003850      THEN WRITE #2, USING CAT,C$,D$,T$,PAGE%
003860      ELSE GOTO FOUT
003870      GOTO ALRITE
003880  FOUT: WRITE #2, USING CAT1,C$,SBTITLE$,D$,T$,PAGE%
003890      IF SBTTTL1$<>"
003900      THEN WRITE #2, USING CAT7,C$,SBTTTL1$
003910  ALRITE: WRITE #2, USING CAT2, C$, FILE$, LIB$, IVOL$
003920      WRITE #2, USING CAT3, C$
003930      WRITE #2, USING CAT4, C$
003940      WRITE #2, USING CAT5, C$
003950      WRITE #2, USING CAT3, C$
003960      RETURN
003970
003980  DOG0:###      !      #####
003990      #####      !
004000  DOG: ###
004010  DG:###      !!! END OF LISTING !!!
004020  DOG1:###      !-----!
004030      -----!
004040  DOG2:###      !
004050      !
004060  DOG3:###      !
004070      *      !
004080  DOG4:###      !
004090      #####      !
004100  DOG5:###      !
004110      !
004120  DOG6:###      !

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 15:41 11/08/84 Page 8  
Input File is CRES@CWP in Library CRES@SRC on Volume CRES02

```

004130      :                               RT
004140 DOG7:###      :      INPUT VOLUME: #####      : RT
004150      :                               RT
004160 DOG8:###      :      STARTING WORD: #####      : RT
004170 #####      :                               RT
004180 DOG9:###      :      ENDING WORD: #####      : RT
004190 #####      :                               RT
004200 DOG10:###      :      #####      : RT
004210 #####      :                               RT
004220 CAT:### #####      : RT
004230      PAGE #####      : RT
004240 CAT1:### TITLE OF LIST : #####      : RT
004250 ##### PAGE #####      : RT
004260 CAT7:### #####      : RT
004270 CAT2:### Input File is ##### in Library ##### on Volume ###: RT
004280 ###      : RT
004290 CAT3:###=====      : RT
004300 =====      : RT
004310 CAT4:### word      Part of      Past      To : RT
004320 Be      Negative      : RT
004330 CAT5:###      Speech      Participle      : RT
004340 CAT6:### # #####      #      #: RT
004350 #      : RT
004360 CAT9:### #####      : RT
004370 CAT8:### + = ROOT WORD      : RT
004380 RAT:### BOLDFAE = Root Word      : RT
004390 F .Tl:### BOLDFAE      : RT
004400      : RT
004410 /* THIS FUNCTION PRINTS OUT ERROR MESSAGES AND */      : RT
004420 /* SETS UP FLASHING FAC'S IN APPROPRIATE PLACES */      : RT
004430 DEF FN'7(MSG$,B$,C$,D$)      : RT
004440 PRINT      : RT
004450 MS=HEX(94)      : RT
004460 GS=B$      : RT
004470 LS=C$      : RT
004480 VS=D$      : RT
004490 RETURN      : RT
004500      : RT
004510 EXIT: CLOSE #1      : RT
004520 OUT: /* RETURNING TO COMMON WORD LIST MENU */      JM

```

Print Common Words By Attribute (CRES@CPA)

CRES@CPA in library CRES@SRC is the BASIC language program to print the Common Word List by attribute. The object language version is CRES@CPA in library CRES@SAV.

This program first allows the user to specify the word list to be printed and the attributes, (part of speech, past participle, form of the verb "to be", negative or root word) to be included. See table 1 for a definition of the parts of speech.

If a particular part of speech is selected which also appears in combination with some other part(s) of speech, all combinations which contain the originally specified part of speech will be included. (For example, selecting verb will produce a list of verbs, adverb/verbs, noun/verbs, and adjective/verbs.) More than one attribute can be selected, and all words containing any attribute selected will be included in the printout. If the root word attribute is selected in combination with another attribute, then only root words possessing that attribute will be printed.

The user can select a subtitle of up to two lines to be printed on each page of the output listing. The output listing can be sent to either the terminal or the printer, it can encompass the entire output list generated, or any portion of the list, and the number of lines per page can be specified.

A complete program listing follows. Appendix C contains a sample flow chart.

Technical Report 86-001

THIS PAGE INTENTIONALLY LEFT BLANK.

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 15:12 11/08/84 Page 1  
Input File is CRES@CPA in Library CRES@SRC on Volume CRES02

```

000010 /*This is the CRES COMMON WORD LIST PRINT BY ATTRIBUTE PROGRAM */ JM
000020 /* */
000030 /* AUTHOR: Rose Tank */ JM
000040 /* Date: January 12, 1984 */ JM
000050 /* */
000060
000070 /* The following code sets up a screen, which asks for the */
000080 /* name of the file containing the common word list */
000090 /* to be printed. It also provides the opportunity to change */
000100 /* the library and volume names which are set to the appro- */
000110 /* priate defaults through the "Extract" system routine. */
000120
000130
000140 DIM WORD$24,ERR1$60,CODE$1,MSG$50,F$40,G$1,CMSC$255,MSG2$40
000150 DIM F1$20,IOBUFF$64,RVOL$06,PFILE$08,FILE$8,ILIB$8,IVOL$6 JM
000160 DIM ENTER$5,PF14$4,PF15$4,PF16$4,ER$79,RLIB$8,E2$37 JM
000170 DIM HEADER$51,L$1,V$1,B$1,C$2,D$8,M$1,MSK$2,MSG1$50
000180 DIM SBTITLE$20,A$(20)15,SBTITL1$20,STARTPH$20,ENDPH$20,TP$8
000190 DIM PSPCTIT$15,OTHERS$16,T$(20)1,X$1,SLIB$8,SVOL$6 JM
000200
000210 A$(1)= "adjective"
000220 A$(2)= "verb"
000230 A$(3)= "conjunct."
000240 A$(4)= "adj/verb"
000250 A$(5)= "noun"
000260 A$(6)= "pronoun"
000270 A$(7)= "interject"
000280 A$(8)= "noun/verb"
000290 A$(9)= "adverb"
000300 A$(10)= "preposit"
000310 A$(11)= "article"
000320 A$(12)= "adv/verb"
000330 A$(13)= "noun/adj" RT
000340 A$(14)= "past participle" RT
000350 A$(15)= "to be" RT
000360 A$(16)= "negative" RT
000370 A$(17)= "root word" RT
000380
000390 F1$=HEX(00140001000100010003A00100010003000B)
000400 ENTER$="ENTER": PF14$="PF14":PF15$="PF15":PF16$="PF16" RT
000410 ATTRIBUTES$="ATTRIBUTES" RT
000420 PSPCTIT$="Parts of Speech"
000430 OTHERS$="Other Attributes"
000440 X$="X" RT
000450 F$="Filename not found on (library/volume).
000460 ERR1$="I/O ERROR - Please respecify a COMMON word list."
000470 HEADER$="CRES PRINT BY ATTRIBUTE SUBSYSTEM"
000480 E2$=" Please specify a COMMON word list." JM
000490 FILE$="Common Word List"
000500 CALL "EXTRACT" ADDR("RV",RVOL$,"RL",RLIB$,"IL",ILIB$,"IV",IVOL$, ! JM
000510 "SL",SLIB$,"SV",SVOL$) JM
000520 FILE$="CMNWDLST"
000530 G$=HEX(81) JM
000540 MSG$=" ":L$=HEX(81)
000550 V$=HEX(81): M$=HEX(8D)
000560
000570 FILMNU:

```

```

000580 ACCEPT
000590 AT (01,01),
000600 "CRES: Print Common Words by Attribute Subsystem", JM
000610 AT (01,57),
000620 "(File Selection Screen)",
000630 AT (02,01),
000640 "=====
000650 =====",
000660 AT (04,17),
000670 "Welcome to the",
000680 AT (04,28), FAC(HEX(85)),
000690 HEADER$,
000700 AT (07,14),
000710 "Please specify the",
000720 AT (07,33), FAC(HEX(84)),
000730 FILE$,
000740 AT (07,50),
000750 "you want to print,", JM
000760 AT (08,18),
000770 "if the default values are not satisfactory.",
000780 AT (13,30),
000790 "File =",
000800 AT (13,43),FAC(G$),FILE$, CH(08),
000810 RANGE ( ":", " ", "cccccccccccc"),
000820 AT (14,30),
000830 "Library =", JM
000840 AT (14,43),FAC(L$), ILIB$, CH(08),
000850 RANGE ( ":", " ", "cccccccccccc"),
000860 AT (15,30),
000870 "Volume =",
000880 AT (15,43),FAC(V$), IVOL$, CH(06),
000890 RANGE ( ":", " ", "cccccccccccc"),
000900 AT (18,24), FAC(M$), MSG$,
000910 AT (20,30), "Press", RT
000920 AT (20,36),FAC(HEX(84)), , RT
000930 ENTER$, RT
000940 AT(20,42), RT
000950 "to continue",
000960 AT (21,17), "Press", JM
000970 AT (21,23), FAC(HEX(84)), JM
000980 PF16$,
000990 AT (21,28), JM
001000 "to return to the Common List Word Menu", JM
001010 ON (BIN(16)&BIN(32)) GOTO OUT,OUT RT
001030
001040 M$=HEX(8D)
001041 JM
001050 /* SET UP THE USER FILE BLOCK */
001060
001070 SELECT #1,"COMMON",VARC,INDEXED,RECSIZE=40,KEYPOS=1,KEYLEN=20, :
001080 IOERR GOTO IOERR1
001090
001100 SELECT #2,"PRTFILE",PRINTER,RECSIZE=80,IOERR GOTO IOERR1
001110
001120 /* SEE IF THE SPECIFIED FILE EXISTS */
001130
001140 CALL "LIMITS"(FILE$,ILIB$,IVOL$,CODE$,MSG$) JM

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 15:12 11/08/84 Page 3  
Input File is CRESECPA in Library CRESESRC on Volume CRE502

```

001150 IF CODE$="Y" THEN ZZZ
001160 IF CODE$="N" THEN GOSUB'7(F$,HEX(91),HEX(81),HEX(81))
001170 IF CODE$="V" THEN GOSUB'7(STR(MSG$,3),HEX(81),HEX(81),HEX(91))
001180 IF CODE$="L" THEN GOSUB'7(STR(MSG$,3),HEX(81),HEX(91),HEX(81))
001190 IF CODE$="E" THEN GOSUB'7(STR(MSG$,3),HEX(81),HEX(81),HEX(81))
001200 GOTO FILMNU
001210
001220 , * IF WE FALL HERE, OUR FILE EXISTS - BUT IS IT COMMON WORDS? */ JM
001230
001240 ZZZ: CALL "READPDR" ADDR(FILE$,ILIB$,IVOL$,0%,"RS",RAG%,RET%) JM
001250 IF RAG%=40 THEN ZZ1 JM
001260 GOSUB'7(E2$,HEX(91),HEX(81),HEX(81)) JM
001261 GOTO FILMNU JM
001270 ZZ1: OPEN NOGETPARM #1,SHARED, FILE= FILE$, LIBRARY = ILIB$, : JM
001280 VOLUME = IVOL$
001290
001300 READ #1,IOBUFF$, DATA GOTO IOERR2
001310 IF STR(IOBUFF$,,13)=" COMMON WORDS" THEN YES
001320 CLOSE #1
001330 M$=HEX(94)
001340 MSG$='ERRR1$
001350 GOTO FILMNU
001360 YES:
001370 X$="X": RTFL=0: FOR I=1 TO 17: I$(I)=" ":NEXT I RT
001380
001390 ACCEPT : RT
001400 AT (01,03), : RT
001410 "CRES: Print Common Words by Attribute Subsystem (Attribute Sele: JM
001420 ction Screen)", : RT
001430 AT (02,03), : RT
001440 "===== : RT
001450 =====", : RT
001460 AT (04,03), : RT
001470 " Place an", : RT
001480 AT (04,14),FAC(HEX(85)),X$, CH(01), : RT
001490 AT (04,16), : RT
001500 "beside the attribute or combination of attributes to be printed. : RT
001510 ", : RT
001520 AT (05,03), : RT
001530 "If you choose a part of speech which is included in a combinatio: RT
001540 n, then the", : RT
001550 AT (06,03), : RT
001560 "combination will also be printed. (e.g. If adverb is selected th: RT
001570 en adv/verbs", : RT
001580 AT (07,03), : RT
001590 "will printed as well as the adverbs.) If the root word option i: RT
001600 s selected in", : RT
001610 AT (08,03), : RT
001620 "combination with another attribute, then only root words with th: RT
001630 at attribute", : RT
001640 AT (09,03), : RT
001650 "will be printed.", : RT
001660 AT (11,37), FAC(HEX(AD)), ATTRIBUTES$ , CH(10), : RT
001670 AT (13,14), FAC(HEX(AC)), PSPCFIT$ , CH(15), : RT
001680 AT (13,55), FAC(HEX(AC)), OTHERS$ , CH(16), : RT
001690 AT (14,05), : RT
001700 "adjective", : RT

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 15:12 11/08/84 Page 4  
Input File is CRESECPA in Library CRES\$SRC on Volume CRES02

001710	AT (14,18), FAC(HEX(81)),	T\$(1)	, CH(01),	: RT
001720	AT (14,24),			: RT
001730	"noun/verb",			: RT
001740	AT (14,36), FAC(HEX(81)),	T\$(8)	, CH(01),	: RT
001750	AT (14,55),			: RT
001760	"past participle",			: RT
001770	AT (14,71), FAC(HEX(81)),	T\$(14)	, CH(01),	: RT
001780	AT (15,05),			: RT
001790	"verb",			: RT
001800	AT (15,18), FAC(HEX(81)),	T\$(2)	, CH(01),	: RT
001810	AT (15,24),			: RT
001820	"adverb",			: RT
001830	AT (15,36), FAC(HEX(81)),	T\$(9)	, CH(01),	: RT
001840	AT (15,55),			: RT
001850	"to be",			: RT
001860	AT (15,71), FAC(HEX(81)),	T\$(15)	, CH(01),	: RT
001870	AT (16,05),			: RT
001880	"conjunction",			: RT
001890	AT (16,18), FAC(HEX(81)),	T\$(3)	, CH(01),	: RT
001900	AT (16,24),			: RT
001910	"preposition",			: RT
001920	AT (16,36), FAC(HEX(81)),	T\$(10)	, CH(01),	: RT
001930	AT (16,55),			: RT
001940	"negative",			: RT
001950	AT (16,71), FAC(HEX(81)),	T\$(16)	, CH(01),	: RT
001960	AT (17,05),			: RT
001970	"adj/verb",			: RT
001980	AT (17,18), FAC(HEX(81)),	T\$(4)	, CH(01),	: RT
001990	AT (17,24),			: RT
002000	"article",			: RT
002010	AT (17,36), FAC(HEX(81)),	T\$(11)	, CH(01),	: RT
002020	AT (17,55),			: RT
002030	"root word",			: RT
002040	AT (17,71), FAC(HEX(81)),	T\$(17)	, CH(01),	: RT
002050	AT (18,05),			: RT
002060	"noun",			: RT
002070	AT (18,18), FAC(HEX(81)),	T\$(5)	, CH(01),	: RT
002080	AT (18,24),			: RT
002090	"adv/verb",			: RT
002100	AT (18,36), FAC(HEX(81)),	T\$(12)	, CH(01),	: RT
002110	AT (19,05),			: RT
002120	"pronoun",			: RT
002130	AT (19,18), FAC(HEX(81)),	T\$(6)	, CH(01),	: RT
002140	AT (19,24),			: RT
002150	"noun/adj",			: RT
002160	AT (19,36), FAC(HEX(81)),	T\$(13)	, CH(01),	: RT
002170	AT (20,05),			: RT
002180	"interjection",			: RT
002190	AT (20,18), FAC(HEX(81)),	T\$(7)	, CH(01),	: RT
002200	AT (21,24), FAC(M\$),	MSG\$	, CH(45),	: RT
002210	AT (23,30),			: RT
002220	"Press",			: RT
002230	AT (23,36), FAC(HEX(85)),	ENTER\$	, CH(05),	: RT
002240	AT (23,42),			: RT
002250	"to continue",			: RT
002260	AT (24,17),			: JM
002270	"Press",			: RT

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 15:12 11/08/84 Page 5  
Input File is CRES@CPA in Library CRES@SRC on Volume CRES02

```

002280      AT (24,23), FAC(HEX(84)),      PF16$      , CH(04), ! JM
002290      AT (24,28),                      ! JM
002300 "to return to the Common Word List Menu",      ! JM
002310 KEYS(BIN(0)&BIN(16)&BIN(32)),          ! JM
002320 ON (BIN(16)&BIN(32)) GOTO EXIT,EXIT      JM
002330
002340 MSG$=" ":MS$=HEX(8D): K=0
002350 FOR I = 1 TO 17                      RT
002360 PSN=POS(" X"=T$(I))
002370 IF PSN=0 THEN INPTER
002380 IF PSN=2 THEN K=1
002390 NEXT I
002400 IF K=1 THEN 2460                      JM
002410 IF K=0 AND T$(17)="X" THEN RTFL=2 ELSE INPTER      RT
002420 GOTO 2460                      JM
002430 INPTER: MSG$="Please use only 'X' to mark attributes."
002440 MS$=HEX(94): GOTO YES
002450
002460 IF T$(1)<>"X" THEN 2480      /*If adj was marked then      */ JM
002470      T$(4)="X": T$(13)="X"      /*also print adj/verb & noun/adj*/ RT
002480 IF T$(2)<>"X" THEN 2510      /*If verb was marked then also */ JM
002490      T$(4)="X": T$(8)="X": T$(12)="X"/*print adj/verb, noun/verb*/ RT
002500      /* & adv/verb      */ RT
002510 IF T$(5)<>"X" THEN 2530      /*If noun was marked then also */ JM
002520      T$(8)="X": T$(13)="X"      /*print noun/verb & noun/adj */ RT
002530 IF T$(9)="X" THEN T$(12)="X"      /*If adv was marked then also */ RT
002540      /*print adv/verb      */ RT
002550      RT
002560 FOR I =14 TO 16: IF T$(I)="X" THEN T$(I)="*" ELSE T$(I)="$":NEXT I RT
002570 IF K=1 AND T$(17)="X" THEN RTFL=1      RT
002580 /* This section of code generates another screen, which asks      */ JM
002590 /* for the particular word interval and the corresponding      */ JM
002600 /* print operation. The user is given the option of returning */
002610 /* to the previous screen to select new attributes to print, */ JM
002620 /* or of returning to the common word list menu.      */ JM
002630
002640
002650 ER$ = " "
002660 CWMENU:
002670 WORD$ = " "
002680 LPP$      =50
002690 PF15$      ="PF15"
002700 PF16$      ="PF16"
002710 SBTITLE$   =" "
002720 SBTITL1$   =" "
002730 BALONEY:
002740 TP$        ="TERMINAL"
002750 STARTPH$    ="FIRST"
002760 ENDPH$     ="LAST"
002770 PRINTSCR:
002780 ACCEPT
002790      AT (01,03),                      :
002800 "CRES: Print Common Words by Attribute Subsystem",      JM
002810      AT (01,66),                      :
002820 "(Print Screen)",                      :
002830      AT (02,03),                      :
002840 "=====

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 15:12 11/08/84 Page 6  
Input File is CRES@CPA in Library CRES@SRC on Volume CRES02

```

002850 =====,
002860     AT (04,13),
002870 "To print a range of lines, fill in the information below.",
002880     AT (06,02), FAC(HEX(85)),
002890 ER$,
002900     AT (08,22),
002910 "Subtitle of list =",
002920     AT (08,42), FAC(HEX(81)),SBTITL$,CH(20),
002930     AT (09,42), FAC(HEX(81)),SBITL$,CH(20),
002940     AT (12,22),
002950 "Starting word =",
002960     AT (12,42),FAC(HEX(81)), STARTPH$, CH(20),
002970     RANGE ( "!", " ", "#####"),
002980     AT (13,22),
002990 "Ending word =",
003000     AT (13,42), FAC(HEX(81)), ENDPH$, CH(20),
003010     RANGE ( "!", " ", "#####"),
003020     AT (14,22),
003030 "Lines per page =",
003040     AT (14,42), FAC(HEX(82)), LPP$, PIC(##),
003050     AT (17,22),
003060 "Print at (TERMINAL/PRINTER) =",
003070     AT (17,53), FAC(HEX(81)), TP$, CH(08),
003080     AT (22,32), JM
003090 "Press",
003100     AT (22,38), FAC(HEX(85)), ENTER$, CH(05), JM
003110     AT (22,44), JM
003120 "to print",
003130     AT (23,24), JM
003140 "Press",
003150     AT (23,30),FAC(HEX(85)),PF15$,CH(04), JM
003160     AT (23,35), "to select new attributes", JM
003170     AT (24,19),
003180 "Press",
003190     AT (24,25), FAC(HEX(84)), PF16$, CH(04),
003200     AT (24,30),
003210 "to return to Common Word List Menu",
003220 KEYS(BIN(0)&BIN(16)&BIN(32)&BIN(15)&BIN(31)), JM
003230 ON (BIN(16)&BIN(32)&BIN(15)&BIN(31)) GOTO EXIT,EXIT,YES,YES JM
003240 JM
003250     IF STR(TP$,1,1) = "P" THEN NEXT11
003260     IF STR(TP$,1,1) <> "T" THEN TERMPR
003270 NEXT11:     IF LPP$ < 51 THEN NEXT13
003280     ER$="Lines per page' can be at most 50."
003290     GOTO PRINTSCR
003300
003310 IOERR1: PRINT
003320     PRINT BELL,HEX(94),ERR1$
003330     CLOSE #1
003340     FOR I=1 TO 200000 :NEXT I
003350     GOTO FILMNU
003360
003370 TERMPR:
003380 ER$="Print at ' has only legal values TERMINAL or PRINTER."
003390     GOTO PRINTSCR
003400
003410 REX: ER$="          'Starting word' comes after 'Ending word'."

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 15:12 11/08/84 Page 7  
Input File is CRES@CPA in Library CRES@SRC on Volume CRES02

```

003420      GOTO PRINTSCR
003430
003440
003450 IOERR2:  PRINT
003460          PRINT HEX(94),"Error occurred on file "; FILE$
003470          PRINT HEX(94),"Error code = "; PS(#1)
003480          STOP "Touch ENTER to return to menu."
003490          GOTO CWMENU
003500
003510 NEXT13:  IF STARTPH$="FIRST" OR ENDPH$="LAST" THEN FORD
003520          IF STARTPH$>ENDPH$ THEN REX
003530
003540 /* Now it's time to generate our print file name */
003550
003560 FORD:    DAT$= DATE
003570          D$= STR(DAT$,3,2)&"/"&STR(DAT$,5,2)&"/"&STR(DAT$,1,2)
003580          T$= STR(TIME,1,2)&":"&STR(TIME,3,2)
003590
003600          PFILE$="CAPRIB##"
003610          I%=0
003620 NEXT14:  CONVERT I% TO STR(PFILE$,7,2),PIC(00)
003630          I%=I%+1
003640          CALL "LIMITS" (PFILE$,SLIB$,SVOL$,CODE$,MSG1$)      JM
003650          IF CODE$="Y" THEN NEXT14
003652          IF CODE$="E" THEN ER$ = STR(MSG$,3)
003654          IF CODE$="E" THEN PRINTSCR
003660 DISPLAY                                     : JM
003670          AT (01,03),
003680 "CRES: Print Common Words by Attribute Subsystem",           : JM
003690          AT (01,61),
003700 "(Generation Screen)",                                       : JM
003710          AT (02,03),
003720 "=====
003730 =====",
003740          AT (12,23),
003750 "Output file now being generated!"                             JM
003760          OPEN NOGETPARM #2,OUTPUT,FILE=PFILE$,LIBRARY=SLIB$, : JM
003770          VOLUME=SVOL$,SPACE=15000                             JM
003780          GOSUB TTL PGE
003790          IF STARTPH$ = "FIRST" THEN STARTPH$ = ALL(HEX(00))
003800          IF ENDPH$ = "LAST" THEN ENDPH$ = ALL(HEX(FF))
003810          PAGE% = 1
003820          RECS% = 1
003830          READ #1, KEY>=STARTPH$,IOBUFF$
003840          IF STR(IOBUFF$,1,20)>ENDPH$ THEN HOME
003850          IF STR(IOBUFF$,1,20)=" COMMON WORDS" THEN AGAN
003860 KNOT:    IF RTFL<>1 THEN 3880                                JM
003870          IF STR(IOBUFF$,26,1)<>"*" THEN AGAN
003880          PSN= POS("J 3jNOInAPBad"= STR(IOBUFF$,21,1))         RT
003890          IF T$(PSN)="X" THEN 3950                               JM
003900          IF T$(14)=STR(IOBUFF$,22,1) THEN 3950                 JM
003910          IF T$(15)=STR(IOBUFF$,24,1) THEN 3950                 JM
003920          IF T$(16)=STR(IOBUFF$,23,1) THEN 3950                 JM
003930          IF RTFL=2 AND STR(IOBUFF$,26,1)="*" THEN 3950         JM
003935          IF T$(17)="X" AND STR(IOBUFF$,26,1)="*" THEN 3950    JM
003940          GOTO AGAN
003950          IF RECS% = 1 THEN GOSUB HEADER

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 15:12 11/08/84 Page 8  
Input File is CRES@CPA in Library CRES@SRC on Volume CRES02

```

003960      IF STR(TP$,1,1)="I" THEN BUG
003970      WRITE #2, USING IMG6,C$, " ",STR(IOBUFF$,1,20),
003980      AS(PSN),STR(IOBUFF$,22,1),STR(IOBUFF$,24,1),
003990      STR(IOBUFF$,23,1)
004000      IF (STR(IOBUFF$,26,1)<>"")THEN BAG
004010      IF RTFL<>0 THEN BAG
004020      C$=HEX(0000)
004030      WRITE #2, USING IMG9, C$,STR(IOBUFF$,1,20)
004040      WRITE #2, USING IMG9, C$,STR(IOBUFF$,1,20)
004050      C$=HEX(0001)
004060      GOTO BAG
004070 BUG: IF STR(IOBUFF$,26,1)="*" THEN STR(IOBUFF$,26,1)="+"
004080      WRITE #2, USING IMG6,C$,STR(IOBUFF$,26,1),STR(IOBUFF$,1,20),
004090      AS(PSN),STR(IOBUFF$,22,1),STR(IOBUFF$,24,1),
004100      STR(IOBUFF$,23,1)
004110 BAG: RECS%=RECS%+1
004120      IF RECS%>LPP% THEN RECS%=1
004130      IF RECS%=1 THEN PAGE%=PAGE%+1
004140 AGAN: READ #1, IOBUFF$,EOD GOTO HOME
004150      IF STR(IOBUFF$,1,20)>ENDPH$ THEN HOME
004160      GOTO KNOT
004170 HOME: IF RECS%=1 THEN RECS%=51
004180      WRITE #2, USING TIT, C$
004190      RECS%=RECS%+2
004200      WRITE #2, USING DG, C$
004210      FOR I = RECS% TO LPP%+1
004220      WRITE #2, USING TIT, C$
004230      NEXT I
004240      IF STR(TP$,1,1)="P" THEN HER1
004250      WRITE #2, USING IMG8, C$
004260      GOTO HER
004270 HER1: IF RTFL<>0 THEN HER: WRITE #2, USING RT, HEX(0002)
004280      C$=HEX(0000)
004290      FOR I=1 TO 2
004300      WRITE #2, USING RT1, C$
004310      NEXT I
004320      C$=HEX(0001)
004330 HER: CLOSE #2
004340      IF STR(TP$,1,1) ="P" THEN MESS
004350      CLOSE WS
004360      MSG2$ = "CANCEL Processing"
004370      MSK$=HEX(00)
004380      CMSG$ = " "
004390      CALL "LINK" ADDR("DISPFILE","P",RLIB$,RVOL$,3%,PFILE$,
004400      SLIB$,SVOL$, "C",MSG2$,27%, "H",MSK$,CMSG$,255%,C$,R%)
004410      ER$ = " "
004420      IF C%<>0 OR R%<>0
004430      THEN ER$ = STR(CMSG$,141)
004440      GOTO CWMENU
004450
004460 MESS: ER$=" File "&PFILE$& " in Library "&SLIB$ & " on " &
004470      SVOL$ & " generated."
004480      GOTO BALONEY
004490
004560 PTL_PGE:
004570      C$=HEX(8001)
004580      WRITE #2, USING TIT, C$

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 15:12 11/08/84 Page 9  
Input File is CRESECPA in Library CRESESRC on Volume CRES02

```

004590      C$ = HEX(0012)
004600      WRITE #2, USING TIT, C$
004610      C$ = HEX(0001)
004620      WRITE #2, USING TIT1, C$
004630      WRITE #2, USING TIT2, C$
004640      WRITE #2, USING TIT0, C$,D$,T$
004650      WRITE #2, USING TIT3, C$
004660      WRITE #2, USING TIT2, C$
004670      WRITE #2, USING TIT1, C$
004680      WRITE #2, USING TIT2, C$
004690      IF SBTITLE$ = "                " THEN GERRY
004700      WRITE #2, USING TIT4, C$, SBTITLE$
004710      IF SBTITL1$<>"                " THEN WRITE #2, USING :
004720      TIT10, C$, SBTITL1$
004730      WRITE #2, USING TIT2, C$
004740      GERRY: WRITE #2, USING TIT5, C$, FILE$
004750      WRITE #2, USING TIT2, C$
004760      WRITE #2, USING TIT6, C$, ILIB$                JM
004770      WRITE #2, USING TIT2, C$
004780      WRITE #2, USING TIT7, C$, IVOL$
004790      WRITE #2, USING TIT2, C$
004800      WRITE #2, USING TIT8, C$, STARTPH$
004810      WRITE #2, USING TIT2, C$
004820      WRITE #2, USING TIT9, C$, ENDPH$
004830      FOR I= 1 TO 17
004840          IF T$(I)="X" OR T$(I)="*" THEN FIRSTAT
004850      NEXT I
004860      FIRSTAT: WRITE #2,USING TIT2,C$: WRITE #2,USING TIT11,C$,A$(I)
004870      FOR K= I+1 TO 17
004880          IF T$(K)="X" OR T$(K)="*" THEN WRITE #2, USING TIT12, C$,A$(K)
004890      NEXT K
004900      WRITE #2, USING TIT2, C$
004910      WRITE #2, USING TIT1, C$
004920      RETURN
004930
004940      HEADER:
004950          IF PAGE%=1 THEN FIRST
004960          C$ = HEX(0001)
004970          IF STR(TP$,1,1)="P" THEN FRST1
004980          WRITE #2, USING TIT, C$
004990          WRITE #2, USING IMG8, C$
005000          GOTO FIRST
005010      FRST1: IF RTFL<>0 THEN FIRST
005020          WRITE #2, USING RT, HEX(0002)
005030          C$=HEX(0000)
005040          FOR I=1 TO 2
005050              WRITE #2, USING RPI, C$
005060          NEXT I
005070      FIRST: C$ = HEX(8001)
005080          WRITE #2, USING TIT, C$
005090          C$ = HEX(0001)
005100          IF SBTITLE$ = "                " :
005110              THEN WRITE #2, USING IMG,C$,D$,T$,PAGE% :
005120              ELSE GOTO FOUT :
005130          GOTO ALRITE
005140      FOUT: WRITE #2, USING IMG1,C$,SBTITLE$,D$,T$,PAGE%
005150          IF SBTITL1$<>"                " :

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 15:12 11/08/84 Page 10  
Input File is CRESECPA in Library CRESESRC on Volume CRESE02

```

005160      THEN WRITE #2, USING IMG7,C$,S$TITL1$
005170 ALRITE: WRITE #2, USING IMG2, C$, FILE$, ILIB$, IVOL$      JM
005180      WRITE #2, USING IMG3, C$
005190      WRITE #2, USING IMG4, C$
005200      WRITE #2, USING IMG5, C$
005210      WRITE #2, USING IMG3, C$
005220      RETURN
005230
005240 TIT0:###      :      #####
005250      #####      :
005260 TIT: ###
005270 DG:###      !!! END OF LISTING !!!
005280 TIT1:###      :-----!
005290 -----!
005300 TIT2:###      :
005310      :
005320 TIT3:###      :      ***** COMMON WORD LIST *****
005330 *      :
005340 TIT4:###      :      TITLE OF LIST: #####!
005350 #####      :
005360 TIT5:###      :      INPUT FILE:      #####
005370      :
005380 TIT6:###      :      INPUT LIBRARY: #####
005390      :
005400 TIT7:###      :      INPUT VOLUME: #####
005410      :
005420 TIT8:###      :      STARTING WORD: #####!
005430 #####      :
005440 TIT9:###      :      ENDING WORD:      #####!
005450 #####      :
005460 TIT10:###      :      #####!
005470 #####      :
005480 TIT11:###      :      ATTRIBUTES SPECIFIED: #####!
005490 #####      :
005500 TIT12:###      :      #####!
005510 #####      :
005520 IMG:### #####      :
005530      PAGE #####
005540 IMG1:### TITLE OF LIST : #####      :
005550 #####      PAGE #####
005560 IMG7:### #####
005570 IMG2:### Input File is ##### in Library ##### on Volume ###!
005580 ###
005590 IMG3:###=====!
005600 -----!
005610 IMG4:### WORD      Part of      Past      To : RT
005620 Be      Negative
005630 IMG5:###      Speech      Participle      RT
005640
005650 IMG6:### # ##### #      # RT
005660 #
005670 IMG9:### #####
005680 IMG8:### + = ROOT WORD
005690 RT:### BOLDFACE = Root Word
005700 RT1:### BOLDFACE
005710
005720 /* THIS FUNCTION PRINTS OUT ERROR MESSAGES AND */

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 15:12 11/08/84 Page 11  
Input File is CRES@CPA in Library CRES@SRC on Volume CRES02

```
005730 /* SETS UP FLASHING FAC'S IN APPROPRIATE PLACES */
005740 DEF FN'7(MSG$,B$,C$,D$)
005750     PRINT
005760     M$=HEX(94)
005770     G$=B$
005780     L$=C$
005790     V$=D$
005800 RETURN
005810 .
005820 EXIT: CLOSE #1
005830 OUT: /* RETURNING TO COMMON WORD LIST MENU */
```

JM

Technical Report 86-001

THIS PAGE INTENTIONALLY LEFT BLANK.

## TECHNICAL WORD LIST SUBSYSTEM

### Create/Maintain Technical Word List Menu (CRES@SWL)

CRES@SWL in library CRES@PRC is the Procedure Language menu routine that serves as the driver for the Create/Maintain Technical Word List Subsystem. It enables the user to select by PF key among options to create/maintain any technical word list, to print a technical word list, to print a technical word list by attribute, or to return to the CRES Master Menu.

A complete listing follows.

Technical Report 86-001

THIS PAGE INTENTIONALLY LEFT BLANK.

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 16:28 11/13/84 Page 1  
Input File is CRES@SWL in Library CRES@PRC on Volume CRES01

```

*
PROCEDURE CRES SUPPLEMENTARY WORD LIST MENU
*
AUTHOR ROSE TANK
*
DATE 10/07/83
*
Abstract: This procedure is a subsystem of CRES Master Menu
*
(option 2 - Create/Maintain Technical Word
*
Lists).
*
It allows the user to select and run any
*
Technical word list file option.
*
RT 000100
RT 000200
RT 000300
RT 000400
RT 000500
RT 000600
RT 000700
RT 000800
RT 000900
RT 001000
RT 001100
RT 001200
RT 001300
RT 001400
RT 001500
RT 001600
RT 001700
RT 001800
RT 001900
RT 002000
RT 002100
RT 002200
RT 002300
RT 002400
RT 002500
RT 002600
RT 002700
RT 002800
RT 002900
RT 003000
RT 003100
RT 003200
RT 003300
RT 003400
RT 003500
RT 003600
RT 003700
RT 003750
RT 003800
RT 003900
RT 004000
RT 004100
RT 004200
RT 004300
RT 004400
RT 004500
RT 004600
RT 004700
RT 004800
RT 004900
RT 005000
RT 005010
RT 005100
RT 005110
RT 005120
RT 005130

*
DECLARE &USERINPT AS INTEGER INITIAL 0
DECLARE &OPTIONERR AS STRING (79) INITIAL " "
DECLARE &OPTIONERR2 AS STRING (79) INITIAL " "
DECLARE &USERNAME AS STRING (24) INITIAL " "
DECLARE &ERRMSG AS STRING (34) INITIAL " "
DECLARE &RUNLIB AS STRING (8) INITIAL " "
DECLARE &RUNVOL AS STRING (6) INITIAL " "

*
EXTRACT &USERNAME=USERNAME, &RUNVOL=RUNVOL, &RUNLIB=OUTLIB
*
GETUSERINPT: PROMPT PFKEY = &USERINPT CENTER
DIM "CRES: Technical Word List Subsystem (Technical
1 Word Menu Screen)";
DIM "=====
=====
";
DIM " ";
DIM " Use the appropriate",BRIGHT " Pf key", " to select the desi
red operation.";
DIM " ";
DIM " ",LINE "Pf",DIM " ",
LINE
";
DIM " ";
DIM " ",Bright "1", " - Create/Maintain Technical w
ord List";
DIM " ",Bright "2", " - Print Technical Word List";
DIM " ",Bright "3", " - Print Technical Words by At
tribute";
DIM " ";
DIM " ",Bright "16", " - Return to CRES Master Menu"
;
DIM " ";
BRIGHT &OPTIONERR;
BLINK &OPTIONERR2;
*
ASSIGN &ERRMSG = " "
ASSIGN &OPTIONERR = " "
ASSIGN &OPTIONERR2 = " "
*
IF &USERINPT EQ 16 GOTO USEREXIT
IF &USERINPT EQ 32 GOTO USEREXIT
IF &USERINPT EQ 1 GOTO RUN1WL
IF &USERINPT EQ 17 GOTO RUN1WL
IF &USERINPT EQ 2 GOTO RUN1WP
IF &USERINPT EQ 18 GOTO RUN1WP
IF &USERINPT EQ 3 GOTO RUNPBA
IF &USERINPT EQ 19 GOTO RUNPBA

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 16:28 11/13/84 Page 2  
Input File is CRES@SWL in Library CRES@PRC on Volume CRES01

	RT 005200
ASSIGN &OPTIONERR = "I'm sorry, but your selected option	RT 005300
is not currently available."	RT 005400
ASSIGN &OPTIONERR2 = "Please select another topic."	RT 005500
GOTO GETUSERINPT	RT 005600
*	RT 005700
RUNTWL: RUN CRES@TWL IN &RUNLIB ON &RUNVOL	RT 005800
IF RUNTWL EQ 0 GOTO GETUSERINPT	RT 005900
GOTO USEREXIT	RT 006000
*	RT 006100
RUNTWP: RUN CRES@TWP IN &RUNLIB ON &RUNVOL	RT 006110
IF RUNTWP EQ 0 GOTO GETUSERINPT	RT 006120
GOTO USEREXIT	RT 006130
*	RT 006140
RUNPBA: RUN CRES@TPA IN &RUNLIB ON &RUNVOL	RT 006150
IF RUNPBA EQ 0 GOTO GETUSERINPT	RT 006160
GOTO USEREXIT	RT 006170
*	RT 006180
USEREXIT: RETURN	RT 006200
/	

### Create/Maintain Technical Word List (CRES@TWL)

CRES@TWL in library CRES@SRC is the BASIC language program to create/maintain the technical word lists. The object language version is CRES@TWL in library CRES@SAV.

The Technical Word List Subsystem was included to provide for the situation where there is a body of technical terminology which would be considered uncommon in normal usage, but which is commonly encountered when discussing a particular subject. These words can be included on a technical word list which can be used to supplement the Common Word List when analyzing text dealing with that subject. Any words found on the specified technical word list(s) will be considered "common". As many as seven technical word lists can be specified for inclusion in an analysis.

This program first allows the user to specify the technical word list to be created or maintained. The program then enables the user to add, delete, or modify any given word. A provision is made for clearing the protect bytes, if for some reason those bytes have been left set incorrectly.

The only characters accepted in the input word are the letters A through Z, ".", "-", and "'". There can be no embedded blanks. The data record for each word contains information about the word's attributes: the part of speech (see Table 1), the past participle flag, the negative flag, the form of the verb "to be" flag, and the root word flag. A complete data record description is provided in appendix A.

The program provides for a protection override. A protection capability was included to accommodate multiple users on the system. When a word on any of the CRES word lists is being modified by one user, the protection bytes for the word are set to that user's ID. No other user can access that word while the protection bytes are set. If an attempt is made by one user to access a word protected by another user, a message specifying the user and workstation protecting the word is displayed. When the word is released, the protection bytes are automatically cleared. To provide for the possibility of the system going down, a power failure, or an irregular exit leaving a word with its protection bytes still set, the program contains a protection override capability. For a given user to override the protection, either the word had to be protected under that user's ID, or the user protecting the word has to be no longer logged on to the computer.

A complete program listing follows. Appendix C contains a sample flowchart.

Technical Report 86-001

THIS PAGE INTENTIONALLY LEFT BLANK.

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 15:49 10/30/84 Page 1  
Input File is CRES@TWL in Library CRES@SRC on Volume CRES02

```

000010 /* CRES TECHNICAL WORD LIST MAINTENANCE SUBSYSTEM PROGRAM */ RT
000020 /* */ RT
000030 /*          AUTHORS: Rose Tank and John Mears */ RT
000040 /*          Date:   October 12, 1983 */ RT
000050 /* */ RT
000060 /* This program was developed to maintain the technical word */ RT
000070 /* lists, as well as allow for the creation of new lists. */ JM
000080 /* It is designed to insert, delete, and modify any word in */ RT
000090 /* any technical word list. */ JM
000150 /* Each technical word list is similar in composition to the */ JM
000160 /* common word list. This subsystem has been expanded to */ JM
000170 /* handle multiusers by use of protection bytes set in the */ JM
000180 /* record of the word to be modified. There is an override */ JM
000190 /* facility available within the program itself that operates */ JM
000200 /* on the condition that the user's id, which is extracted */ JM
000210 /* from the system, matches the protection bytes. In the case */ JM
000220 /* where the user attempts to modify a word currently con- */ JM
000230 /* trolled by another, a message is given identifying the user */ JM
000240 /* and his/her location -by workstation number. */ RT
000250 /* */ RT
000260 /* The record layout for the technical word list is as follows: */ RT
000270 /* */ RT
000280 /* character position identifier variable name */ RT
000290 /*      1-20      ==> word WORD$ */ RT
000300 /*      21      ==> part of speech PRTOFSPC$ */ RT
000310 /*      22      ==> past participle PPARTFL$ */ RT
000320 /*      23      ==> negative NEGFL$ */ RT
000330 /*      24      ==> to be flag TOBEFL$ */ RT
000340 /*      25      ==> filler */ RT
000350 /*      26      ==> root word ROOTFL$ */ RT
000360 /*      27      ==> Mandatory tech. word MWLFL$ */ RT
000370 /*      29-31  ==> protection byte PROTECT$ */ RT
000380 /*      32-40  ==> filler for future additions */ RT
000390 /* */ RT
000400 /* The words are stored in an indexed file with the key being */ JM
000410 /* the word itself. */ JM
000420 RT
000430 RT
000440 DIM WORD$20,ERR1$50,CODE$1,MSG$50,F$40,G$1,PROTECT$3,WID$52 JM
000450 DIM F1$20,IOBUFF$40,ER1$50,XYZ$1,EXP$30,F2$10,NOEXIST$66, E$1 JM
000460 DIM PFK$4,TAB$3,BTAB$7,FILE$8,USER$3,USERNAME$24,WRKST$3 RT
000470 DIM LIB$8,FIL$19,IVOL$6,PF$3,DEF$7,A5$10,LEG$5,A7$7,MESSG$21 RT
000480 DIM HEADER$50,L$1,V$1,B$1,C$1,D$1,M$1,MOD$15,PF16$4,ER$66 JM
000490 DIM ENTER$5,PF1$3,P$(13)3,PASSWRD$13,X$1,MESING$14 JM
000500 DIM PPARTFL$1,ONEGFL$1,MWLFL$1,ROOTFL$01,TOBEFL$01 RT
000510 DIM PRTOFSPC$3 RT
000520 RT
000530 ENTER$="ENTER" RT
000540 PF1$="PF1": PF16$="PF16" RT
000550 TAB$="TAB" RT
000560 BTAB$="BACKTAB" RT
000570 EXP$="(Definition of Entries Screen)" JM
000580 MESSG$="Parts of Speech Codes" RT
000590 P$(1)="(J)":P$(2)="(j)":P$(3)="(A)":P$(4)="(a)":P$(5)="(B)" RT
000600 P$(6)="(C)":P$(7)="(I)":P$(8)="(N)":P$(9)="(n)":P$(10)="(P)" RT
000610 P$(11)="(O)":P$(12)="(V)":P$(13)="(d)" RT
000620 I1LE5$      ="PART_OF_SPEECH" JM

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 15:49 10/30/84 Page 2  
Input File is CRES@TWL in Library CRES@SRC on Volume CRES02

```

000630 ITLE6$      ="DEFINITION"                      JM
000640                                                    RT
000650 /* The following two lines are used as formats in the packing */ RT
000660 /* and the unpacking of record field values */          RT
000670 F1$=HEX(A014A001A0C1A001A0010001A001A0010001A0030009) RT
000680 F2$=HEX(001CA0030009)                               RT
000690                                                    RT
000700 /* The following code sets up a screen, which asks for the */ RT
000710 /* name of the file containing the technical word list that is */ RT
000720 /* to be modified. It also provides the opportunity to change */ RT
000730 /* the library and volume names which are set to the appro- */ RT
000740 /* priate defaults through the "Extract" system routine. */ RT
000750 F$= "File name not found on (library/volume). "      RT
000760 ERR1$= "I/O ERROR - Please specify a TECHNICAL word list" RT
000770 CALL "EXTRACT" ADDR("IL",LIB$, "IV",IVOL$, "ID",USER$) JM
000780                                                    JM
000790 AGAIN: FILE$=" " ":G$=HEX(81)                        RT
000800 MSG$=" " ":L$=HEX(81)                                  RT
000810 V$=HEX(81); M$=HEX(8D)                                RT
000820 HEADER$      ="CRES TECHNICAL WORD LIST MAINTENANCE SUBSYSTEM" RT
000830 FIL$         ="Technical Word List"                  RT
000840                                                    RT
000850 BLANKS: FILE$=" " JM
000860 FILMNU:                                              RT
000870 ACCEPT                                              : RT
000880 AT (01,01),                                          : RT
000890 "CRES: Technical Word List Maintenance Subsystem", : RT
000900 AT (01,58),                                          : RT
000910 "(File Selection Screen)",                          : RT
000920 AT (02,01),                                          : RT
000930 "===== " : RT
000940 =====, : RT
000950 AT (04,10),                                          : RT
000960 "Welcome to the", : RT
000970 AT (04,25), FAC(HEX(85)); : JM
000980 HEADER$, : RT
000990 AT (07,07),                                          : RT
001000 "Please specify the", : RT
001010 AT (07,26), FAC(HEX(84)), : RT
001020 FIL$, : RT
001030 AT (07,46),                                          : RT
001040 "you want to modify or create.", : RT
001050 AT (13,30),                                          : RT
001060 "File " =, : RT
001070 AT (13,43), FAC(G$), FILE$, CH(08), : RT
001080 RANGE ( ":", " ", "#####"), : RT
001090 AT (14,30),                                          : RT
001100 "Library " =, : RT
001110 AT (14,43), FAC(L$), LIB$, CH(08), : RT
001120 RANGE ( ":", " ", "#####"), : RT
001130 AT (15,30),                                          : RT
001140 "Volume " =, : RT
001150 AT (15,43), FAC(V$), IVOL$, CH(06), : RT
001160 RANGE ( ":", " ", "#####"), : RT
001170 AT (18,24), FAC(M$), MSG$, : RT
001180 AT (20,30), "Press", : RT
001190 AT (20,36), FAC(HEX(H4)), : RT

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 15:49 10/30/84 Page 3  
Input File is CRES@TWL in Library CRES@SRC on Volume CRES02

```

001200 ENTER$, : RT
001210 AT(20,42), : RT
001220 "to continue", : RT
001230 AT (21,15),"Press", : RT
001240 AT (21,21), FAC(HEX(84)), : RT
001250 PF16$, : RT
001260 AT (21,26), : RT
001270 "to return to the Technical Word List Menu", : RT
001280 ON (BIN(16)&BIN(32)) GOTO OUT,OUT : RT
001290 : RT
001300 MSG$=" ": M$=HEX(8C) : RT
001310 : RT
001320 /* SET UP THE USER FILE BLOCK */ : RT
001330 : RT
001340 SELECT #1,"TECHWD",VARC,INDEXED,RECSIZE=40,KEYPOS=1,KEYLEN=20, : RT
001350 IOERR GOTO IOERR1 : RT
001360 : RT
001370 /* SEE IF THE SPECIFIED FILE EXISTS */ : RT
001380 /* Y- the file exists & the program may continue */ : RT
001390 /* N- the file doesn't currently exist, you may create one */ : RT
001400 /* V- the volume supplied doesn't exist or is not mounted */ : RT
001410 /* L- the library supplied doesn't exist in the volume given */ : RT
001420 /* E- there is an I/O error in the file specified */ : RT
001430 : RT
001440 CALL "LIMITS"(FILE$,LIB$,IVOL$,CODE$,MSG$) : RT
001450 IF CODE$="Y" THEN RECCK : RT
001460 IF CODE$="N" THEN CREATE : RT
001470 IF CODE$="V" THEN GOSUB'7(STR(MSG$,3),HEX(81),HEX(81),HEX(91)) : RT
001480 IF CODE$="L" THEN GOSUB'7(STR(MSG$,3),HEX(81),HEX(91),HEX(81)) : RT
001490 IF CODE$="E" THEN GOSUB'7(STR(MSG$,3),HEX(81),HEX(81),HEX(81)) : RT
001500 GOTO FILMNU : RT
001510 : RT
001520 /* The following screen allows the user to create an additional */ : RT
001530 /* technical word list,or respecify the file information entered*/ : RT
001540 CREATE: : RT
001550 ACCEPT : RT
001560 "CRES: Technical Word List Maintenance Subsystem", : RT
001570 AT (01,53), : RT
001580 "(Create File Option Screen)", : RT
001590 AT (02,01), : RT
001600 "===== : RT
001610 =====", : RT
001620 AT (06,08), : RT
001630 "The file you have selected does not currently exist in the libra: RT
001640 ry", : RT
001650 AT (07,08), : RT
001660 "and volume specified. For your convenience, the following are ", : JM
001670 AT (08,08), : RT
001680 "the parameters you submitted:", : JM
001690 AT (11,30), : RT
001700 "File =", : RT
001710 AT (11,40),FAC(HEX(85)),FILE$ , CH(08), : RT
001720 AT (12,30), : RT
001730 "Library=", : RT
001740 AT (12,40), FAC(HEX(85)), LIB$ , CH(08), : RT
001750 AT (13,30), : RT
001760 "Volume =", : RT

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 15:49 10/30/84 Page 4  
Input File is CRES@TWL in Library CRES@SRC on Volume CRES02

```

001770      AT (13,40), FAC(HEX(85)), IVOL$      , CH(06),      : RT
001780      AT (17,15),      : RT
001790 "To create a new file from these parameters, press",      : JM
001800      AT (17,65), FAC(HEX(85)), ENTER$      , CH(05),      : RT
001810      AT (19,26),      : RT
001820 "To select another file, press",      : JM
001830      AT (19,56), FAC(HEX(85)), PF1$      , CH(03),      : RT
001840 KEYS(BIN(0)&BIN(1)&BIN(17)),KEY(OPTION),      : RT
001850 ON (BIN(1)&BIN(17)) GOTO BLANKS,BLANKS      JM
001860      RT
001870 /* Create the new file and insert a header record to identify it */ RT
001880      OPEN NOGETPARM #1, OUTPUT, FILE= FILE$, LIBRARY=LIB$,      : RT
001890      VOLUME= IVOL$, SPACE=1000      RT
001900      IOBUFF$= " TECHNICAL WORDS "& FILE$      RT
001910      WRITE #1, IOBUFF$      RT
001920      CLOSE #1      RT
001930      GOTO ZZ1      JM
001940      RT
001950 /* Check the record size before opening the file to avoid */ JM
001960 /* accidentally causing an error by opening a file with incorrect */ JM
001961 /* record'size. */ JM
001970 RECC: CALL "READFDR" ADDR(FILE$,LIB$,IVOL$,0%,"RS",RSIZ%,RET%) RT
001980      IF RSIZ%<40 THEN ZZ2      JM
001990 ZZ1: OPEN NOGETPARM #1,SHARED, FILE= FILE$, LIBRARY = LIB$,      : JM
002000      VOLUME = IVOL$      RT
002010      RT
002070      READ #1, IOBUFF$      JM
002080      IF STR(IOBUFF$,1,16)=" TECHNICAL WORDS" THEN YES      RT
002090      CLOSE #1      JM
002100 ZZ2: M$=HEX(94)      JM
002110      MSG$= ERR1$      RT
002120      GOTO FILMNU      RT
002130 YES: IOBUFF$=" "      RT
002140 /* This section of code generates the second screen, which asks */ RT
002150 /* for the particular word and its corresponding operation - */ JM
002160 /* either insert word, delete word, or modify word. The user is */ RT
002170 /* also at this point given the option of returning to the */ JM
002180 /* Technical Word List Menu. */ JM
002190      RT
002200 WID$="Please select the word:"      RT
002210 WORD$,ER$,ER1$=" "      RT
002220 WORDEQ$="word = "      RT
002230 PFK$=" PF "      RT
002240 OPER$=" Operation"      RT
002250 O$="1":1$="2":1H$="3":Q$="14":R$="15":S$="16"      RT
002260 NOEXIST$="The word you are attempting to modify does not exist"& : RT
002270 " in this list."      RT
002280 X$=HEX(81)      JM
002290 E$=HEX(84)      RT
002300 PASSWRD$=HEX(000102030E0F101112131E1F20)      JM
002310      RT
002320 CWMENU:      RT
002330 ACCEPT      : RT
002340      AT (01,01),"CRES: Technical Word List Maint Subsystem",      : RT
002350      AT (01,49), "(Word & Option Selection Screen)",      : RT
002360      AT (02,01), "=====      : RT
002370 =====",      : RT

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 15:49 10/30/84 Page 5  
Input File is CRES0TWL in Library CRES0SRC on Volume CRES02

```

002380      AT (04,08), FAC(HEX(8C)),      : RT
002390 WID$,      : RT
002400      AT (06,28), FAC(HEX (8C)),      : RT
002410 WORDEQ$,      : RT
002420      AT (06,36), FAC(X$),WORD$,      : JM
002430      AT (08,15), FAC( E$), ER$,      : JM
002440      AT (09,23), FAC( E$), ER1$,      : RT
002450      AT (11,08),      : RT
002460 inue, select the PF key corresponding to the desired operation.", : RT
002470      AT (13,30), FAC(HEX(A5)),      : RT
002480 PFK$,      : RT
002490      AT (13,38), FAC(HEX(AC)),      : RT
002500 OPER$,      : RT
002510      AT (15,31), FAC(HEX(84)),      : RT
002520 O$,      : RT
002530      AT (15,39),      : RT
002540 "INSERT word",      : RT
002550      AT (16,31), FAC(HEX(84)),      : RT
002560 T$,      : RT
002570      AT (16,39),      : RT
002580 "DELETE word",      : RT
002590      AT (17,31), FAC(HEX(84)),      : RT
002600 TH$,      : RT
002610      AT (17,39),      : RT
002620 "MODIFY word",      : RT
002630      AT (19,31), FAC(HEX(84)),      : RT
002640 Q$,      : RT
002650      AT (19,39),      : RT
002660 "Protection Override",      : RT
002670      AT (20,31), FAC(HEX(84)),      : RT
002680 R$,      : RT
002690      AT (20,39),      : RT
002700 "Return to Technical Word List Selection", : RT
002710      AT (21,31), FAC(HEX(84)),      : RT
002720 S$,      : RT
002730      AT (21,39),      : RT
002740 "Return to Technical Word List Menu", : RT
002750 KEYS(PASSWRD$),      : JM
002760      KEY(OPTION)      : JM
002771 IF OPTION=15 OR OPTION=31 THEN NEW      : JM
002772 IF OPTION=16 OR OPTION=32 THEN EXIT      : JM
002780      : RT
002790 FOR I=1 TO LEN(WORD$)
002800 IF POS(".*ABCDEFGHIJKLMNPOQRSTUVWXYZ- "=STR(WORD$,I,1))=0 : JM
002810      THEN ILLERR
002820 NEXT I
002830
002840 E$= HEX(80)      : RT
002850 ER$,ER1$=" "      : RT
002860      /* Check for confirmation of deletion */ : RT
002870 IF (OPTION=0)AND(DF=1) THEN DEL      : RT
002880      /* Check for cancellation of deletion */ : JM
002890 IF DF=1 THEN UPDATE      : RT
002900      /* Check for override of protection byte */ : RT
002910 IF (OPTION=14) OR (OPTION=30) THEN OVERRIDE      : RT
002920 IF STR(WORD$,1,1)<>" " THEN REED      : RT
002930      : RT

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 15:49 10/30/84 Page 6  
Input File is CRES0TWL in Library CRES0SRC on Volume CRES02

```

002940 PRINT BELL RT
002950 ER$=" The first character cannot be blank."
002960 ER1$= "Please reenter the appropriate word and PF key." RT
002970 E$= HEX(94) RT
002980 GOTO CWMENU RT
002990 RT
003000 ILLERR:
003010 ER$="Illegal character found in word - please reenter."
003020 E$=HEX(94): GOTO CWMENU RT
003030
003040 OVERRIDE: RT
003050 IF NF=1 AND P%=1 THEN CHK RT
003060 ER$= "Please check if word exists by selecting an option." RT
003070 E$=HEX(84) RT
003080 GOTO CWMENU RT
003090 RT
003100 /* Protection override is confirmed if the protection was set */ RT
003110 /* by a user not currently logged on or if the current user's */ RT
003120 /* id equals the protection bytes. */ JM
003130 CHK: RT
003140 IF (USER$=PROTECT$) OR (WRKST%=-1) THEN REMOVE RT
003150 ER$= "You don't have the clearance to override the protect." RT
003160 ER1$=" " RT
003170 E$= HEX(84) RT
003180 GOTO CWMENU RT
003190 RT
003200 /* Reset the protection bytes to blanks */ JM
003210 REMOVE: PROTECT$= " ":P%=0 RT
003220 READ #1, HOLD, KEY=WORD$,IOBUFF$ RT
003230 $PACK (F=F1$) IOBUFF$ FROM WORD$, PRTOFSPC$, PPARTFL$, : RT
003240 DNEGFL$,TOBEFL$,ROOTFL$,MWLFL$,PROTECT$ RT
003250 REWRITE #1, IOBUFF$, DATA GOTO IOERR2 RT
003260 ER$= " You may now select your option." RT
003270 ER1$= " " RT
003280 E$= HEX(84) RT
003290 GOTO CWMENU RT
003300 RT
003310 IOERR1: PRINT RT
003320 M$=HEX(94) RT
003330 MSG$= ERR1$ RT
003340 CLOSE #1 RT
003350 GOTO FILMNU RT
003360 RT
003370 IOERR2: PRINT RT
003380 PRINT HEX(94),"Error occurred on file "; FILE$ RT
003390 PRINT HEX(94),"Error code = "; FS(#1) RT
003400 STOP "Touch ENTER to return to menu." RT
003410 GOTO CWMENU RT
003420 RT
003430 /* Error message is printed if "ENTER" was hit before the option*/ RT
003440 /* was entered */ RT
003450 REED: IF OPTION<>0 THEN SRCH RT
003460 ER$= "Operation missing - Please select a PF key." JM
003470 E$= HEX(94) RT
003480 GOTO CWMENU RT
003490 RT
003500 SRCH: P%=1 RT

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 15:49 10/30/84 Page 7  
Input File is CRES0TWL in Library CRES0SRC on Volume CRES02

```

003510 READ #1,HOLD,KEY=WORD$,IOBUFF$,EOD GOTO NOTFOUND, : RT
003520 DATA GOTO IOERR2 RT
003530 RT
003540 NF=1 /* The word was found */ RT
003550 RT
003560 /* Unpack and check if currently being modified by another */ RT
003570 RT
003580 $UNPACK (F=F1$) IOBUFF$ TO WORD$,PRTOFSPC$,PPARTFL$, : RT
003590 DNEGFL$,TOBEFL$,ROOTFL$,MWLFL$,PROTECT$ RT
003600 IF PROTECT$= " " THEN NOW RT
003610 CALL "FINDUSER" ADDR(PROTECT$,USERNAME$,WRKST%) RT
003620 CONVERT WRKST% TO WRKST$, PIC(###) RT
003630 E$= HEX(94) RT
003640 ER$=" Sorry, "& USERNAME$&" is currently editing " RT
003650 ER1$=" "& WORD$&" "&" at workstation "& WRKST$ RT
003660 REWRITE #1,IOBUFF$ RT
003670 GOTO CWMENU RT
003680 RT
003690 /* Set the protection bytes */ JM
003700 NOW: PROTECT$= USER$ RT
003710 $PACK (F=F1$) IOBUFF$ FROM WORD$, PRTOFSPC$, PPARTFL$, : RT
003720 DNEGFL$,TOBEFL$,ROOTFL$,MWLFL$,PROTECT$ RT
003730 REWRITE #1, IOBUFF$, DATA GOTO IOERR2 RT
003740 RT
003750 /* NOW BRANCH TO THE OPTION SELECTED */ RT
003760 OPT: ON OPTION GOTO INSERT,DELET,MODIFY RT
003770 ON OPTION-16 GOTO INSERT,DELET,MODIFY RT
003780 RT
003790 NOTFOUND: NF=0 /* The word was not found in the list */ RT
003800 GOTO OPT RT
003810 RT
003820 INSERT: RT
003830 /* NF=0 indicates the word doesn't yet exist so go ahead */ RT
003840 /* and insert. NF=1 indicates the word exists. */ RT
003850 RT
003860 IF NF=1 THEN EXIST RT
003870 MOD$= "(Insert Screen)" RT
003871 MESING$="inserting word" JM
003880 MWLFL$,PPARTFL$,PRTOFSPC$,DNEGFL$,ROOTFL$,TOBEFL$=" " RT
003890 IOBUFF$=" ": PROTECT$= USER$ RT
003900 $PACK (F=F1$) IOBUFF$ FROM WORD$, PRTOFSPC$, PPARTFL$, : RT
003910 DNEGFL$,TOBEFL$,ROOTFL$,MWLFL$,PROTECT$ RT
003920 WRITE #1, IOBUFF$, DATA GOTO IOERR2 RT
003930 GOTO INS RT
003940 RT
003950 /* Coming from insert with the word currently existing, */ RT
003960 /* print an error message and erase the protection byte */ RT
003970 EXIST: RT
003980 ER$= "The word '" & WORD$&" already exists in this list." RT
003990 E$= HEX(94) RT
004000 READ #1, HOLD, KEY= WORD$,IOBUFF$ RT
004010 PROTECT$=" " :P%=0 RT
004020 $PACK (F=F1$) IOBUFF$ FROM WORD$, PRTOFSPC$, PPARTFL$, : RT
004030 DNEGFL$,TOBEFL$,ROOTFL$,MWLFL$,PROTECT$ RT
004040 REWRITE #1, IOBUFF$, DATA GOTO IOERR2 RT
004050 GOTO CWMENU RT
004060 RT

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 15:49 10/30/84 Page 8  
Input File is CRES@TWL in Library CRES@SRC on Volume CRES02

```

004070 /* DELET,DEL,NODEL- all pertain to the delete option */ RT
004080 /* DELET- gives the user a chance to cancel the deletion */ RT
004090 DELET: IF NF=0 THEN NODEL RT
004100 DF=1 RT
004110 ER$= " Press ENTER to confirm deletion of "& WORD$ RT
004120 ER1$=" or press PF14 to cancel deletion." JM
004130 PASSWRD$=HEX(000E1E) JM
004140 X$=HEX(84) JM
004150 E$= HEX(94) RT
004160 GOTO CWMENU RT
004170 RT
004180 /* DEL- deletes the entire record */ RT
004190 DEL: READ #1, HOLD,KEY=WORD$,IOBUFF$ RT
004200 DELETE #1 RT
004210 DF=0 RT
004220 ER$= " "&WORD$&" deleted - Please select your next word." RT
004230 WID$=" Updating "&FILE$&" Technical Word List" JM
004240 E$= HEX(8C) RT
004250 WORD$=" " RT
004260 NF=0 JM
004270 PASSWRD$=HEX(000102030E0F101112131E1F20) JM
004280 X$=HEX(81) JM
004290 GOTO CWMENU RT
004300 RT
004310 /* NODEL- print an error message when the word doesn't exist */ RT
004320 NODEL: ER$= "The word "& WORD$ &" does not exist in this list." RT
004330 E$= HEX(94) RT
004340 GOTO CWMENU RT
004350 RT
004360 MODIFY: IF NF=1 THEN GETIT RT
004370 ER$ = NOEXIST$ /* if word not in list, */ RT
004380 E$=HEX(94) /* set up message and fac and */ RT
004390 GOTO CWMENU /* go back up to previous screen */ RT
004400 RT
004410 GETIT: JM
004440 MOD$="(Modify Screen)" RT
004441 MESING$="modifying word" JM
004450 INS: RT
004460 PF$="PF1" RT
004470 DEF$="CURRENT" RT
004480 AS$="ATTRIBUTES" RT
004490 LEG$="LEGAL" RT
004500 A7$="ENTRIES" RT
004510 ER1$=" " RT
004520 XYZ$=HEX(8C) RT
004530 RT
004540 /* This accept statement generates a screen with the attributes */ RT
004550 /* corresponding to each word appropriately labeled for insert */ RT
004560 /* or modification. It also calls a second screen on request if */ RT
004570 /* there is a need for an explanation of the legal entries for */ RT
004580 /* the parts of speech. */ JM
004590 RT
004600 AOK: RT
004610 RT
004620 ACCEPT : RT
004630 AT (01,01), : RT
004640 "CRES: Technical Word List Maintenance Subsystem", : RT

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 15:49 10/30/84 Page 9  
Input File is CRES@TWL in Library CRES@SRC on Volume CRES02

```

004650      AT (01,66), FAC(HEX(8C)),MOD$,      : RT
004660      AT (02,01), "=====-----"      : RT
004670      "=====-----",                    : RT
004680      AT (03,07),                            : RT
004690 "For an explanation of legal entries, press", : JM
004700      AT (03,50),FAC(HEX(84)), PF$,          : RT
004710      AT (03,54),                            : JM
004720 "; otherwise, enter",                    : RT
004730      AT (04,07),                            : RT
004740 "the appropriate attribute(s).",           : RT
004750      AT (06,33),FAC(HEX(AD)), DEF$,         : RT
004760      AT (06,41),FAC(HEX(AD)), A5$,         : RT
004770      AT (06,58),FAC(HEX(AD)), LEG$,        : RT
004780      AT (06,64),FAC(HEX(AD)), A7$,        : RT
004790      AT (08,07),                            : RT
004800 "WORD:",                                    : RT
004810      AT (08,15),FAC(HEX(84)), WORD$,        : RT
004820      RANGE ( ":", ":", "dddddddddd"),      : RT
004830      AT (08,61),                            : RT
004840 "(A-Z.-')",                                : JM
004850      AT (10,07),                            : RT
004860 "PART OF SPEECH:",                        : RT
004870      AT (10,41),FAC(HEX(80)),PRTOFSPC$, CH(01), : RT
004880      AT (10,57),                            : RT
004890 "(ABCDEFGHIJNOPVadjn)",                    : RT
004900      AT (11,07),                            : RT
004910 "PAST PARTICIPLE:",                      : RT
004920      AT (11,41), PPARTFL$,                  , CH(01), : RT
004930      AT (11,57),                            : RT
004940 "(* or NO ENTRY)",                        : RT
004950      AT (12,07),                            : RT
004960 "NEGATIVE:",                              : RT
004970      AT (12,41), DNEGFL$,                  , CH(01), : RT
004980      AT (12,57),                            : RT
004990 "(* or NO ENTRY)",                        : RT
005000      AT (13,07),                            : RT
005010 "TO BE:",                                : RT
005020      AT (13,41), TOBEFL$,                  , CH(01), : RT
005030      AT (13,57),                            : RT
005040 "(* or NO ENTRY)",                        : RT
005050      AT (14,07),                            : RT
005060 "MANDATORY TECHNICAL WORD:",              : RT
005070      AT (14,41), MWLFL$,                  , CH(01), : RT
005080      AT (14,57),                            : RT
005090 "(* or NO ENTRY)",                        : RT
005100      AT (15,07),                            : RT
005110 "ROOT WORD:",                            : RT
005120      AT (15,41), ROOTFL$, CH(01),           : RT
005130      AT (15,57),                            : RT
005140 "(* or NO ENTRY)",                        : RT
005150      AT (17,25),FAC(XYZ$),ER1$,            : RT
005160      AT (20,17),"Use",FAC(HEX(85)),TAB$,"or",FAC(HEX(85)), : RT
005170      BTAB$,"to move between modifiable fields.", : RT
005180      AT (23,30),                            : RT
005190 "Press",                                    : RT
005200      AT (23,36),FAC(HEX(84)),ENTER$,       : RT
005210      AT (23,42),                            : RT

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 15:49 10/30/84 Page 10  
Input File is CRES@TWL in Library CRES@SRC on Volume CRES02

```

005220 "to continue",
005230 AT (24,21),
005240 "Press",AT(24,27),FAC(HEX(84)), PF16$ ,AT(24,32),
005250 "to exit without ",
005251 AT (24,48), FAC(HEX(8C)), MESING$,
005260 KEYS(BIN(0)&BIN(1)&BIN(17)&BIN(16)&BIN(32)),KEY(OPTION),
005270 ON (BIN(16)&BIN(32)) GOTO EXIOM,EXIOM
005280
005310 IF OPTION=1 OR OPTION=17 THEN DEFIN
005320
005330 /* Check for validity of entries */
005340
005350 IF POS("ABCDEFGHIJNOPVadjn="PRTOFSPC$)>0 THEN NEXT1
005360 ER1$ = "Part of Speech value is illegal."
005370 XYZ$ = HEX(94)
005380 GOTO AOK
005390
005400 NEXT1: IF POS(" *="PPARTFL$)>0 THEN NEXT2
005410 ER1$ = "Past Participle value is illegal."
005420 XYZ$ = HEX(94)
005430 GOTO AOK
005440 NEXT2: IF POS(" *="DNEGFL$)>0 THEN NEXT3
005450 ER1$ = " Negative value is illegal."
005460 XYZ$ = HEX(94)
005470 GOTO AOK
005480 NEXT3: IF POS(" *="TOBEFL$)>0 THEN NEXT4
005490 ER1$ = " To Be value is illegal."
005500 XYZ$ = HEX(94)
005510 GOTO AOK
005560 NEXT4: IF POS(" *="MWLFL$)>0 THEN NEXT5
005570 ER1$ = "Mandatory word value is illegal."
005580 XYZ$ = HEX(94)
005590 GOTO AOK
005600 NEXT5: IF POS(" *="ROOTFL$)>0 THEN UPDATE
005610 ER1$ = " Root word value is illegal."
005620 XYZ$ = HEX(94)
005630 GOTO AOK
005640
005650 /* EXIOM is used to abnormally exit the insert mode and */
005660 /* NOCHAN is used to abnormally exit the modify mode */
005670 EXIOM: READ #1, HOLD,KEY=WORD$,IOBUFF$
005680 IF NF=1 THEN NOCHAN
005690 DELETE #1
005700 ER$= " Word "&WORD$&" was not inserted."
005710 GOTO CWMENU
005720
005730 NOCHAN: PROTECT$ = " " :P%=0
005740 $PACK (F=F2$) IOBUFF$ FROM PROTECT$
005750 REWRITE #1,IOBUFF$,DATA GOTO IOERR2
005760 ER$= " No changes were saved."
005770 ER1$=" "
005780 GOTO CWMENU
005790
005800 /* UPDATE removes the protection on the record and writes out */
005810 /* the final data as inputted. */
005820 UPDATE: READ #1, HOLD, KEY=WORD$, IOBUFF$
005830 PROTECT$ = " " :P%=0

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 15:49 10/30/84 Page 11  
Input File is CRES@TWL in Library CRES@SRC on Volume CRES02

```

005840      $PACK (F=F1$) IOBUFF$ FROM WORD$,PRTOFSPC$,PPARTFL$,      : RT
005850      DNEGFL$,TOBEFL$,ROOTFL$,MWLFL$,PROTECT$      RT
005860      REWRITE #1,IOBUFF$,DATA GOTO IOERR2      RT
005870      ER1$= " "
005880      IF NF=0 THEN RITE      /* NF=0 indicates insert mode */      RT
005890      IF DF=1 THEN EXDEL      /* DF=1 indicates delete mode */      JM
005900      WID$= "      Updating "&FILE$&" Technical Word List"      JM
005910 ER$= "      "&WORD$&" updated - Please select next word."      JM
005920      E$=HEX(8C)      RT
005930      WORD$= " "      RT
005940      GOTO CWMENU      RT
005950 RITE:      RT
005960 ER$= "      "&WORD$&" inserted - Please select your next word."      RT
005970      WID$= "      Updating "&FILE$&" Technical Word List"      JM
005980      E$= HEX(8C)      RT
005990      WORD$= " "      RT
006000      GOTO CWMENU      RT
006010      RT
006020 EXDEL: DF=0      RT
006030      ER$ = "      Deletion cancelled."      RT
006040      PASSWRD$=HEX(000102030E0F101112131E1F20)      JM
006050      X$=HEX(81)      JM
006060      GOTO CWMENU      RT
006070      RT
006080 NEW: PRINT PAGE, AT(12,28),HEX(84);"Closing last file"      RT
006090      FOR I= 1 TO 50000:NEXT I      RT
006100      CLOSE #1      RT
006110      GOTO AGAIN      RT
006120      RT
006130 /* THIS FUNCTION PRINTS OUT ERROR MESSAGES AND */      RT
006140 /* SETS UP FLASHING FAC'S IN APPROPRIATE PLACES */      RT
006150 DEF FN'7(MSG$,B$,C$,D$)      RT
006160      PRINT      RT
006170      M$=HEX(94)      RT
006180      G$=B$      RT
006190      L$=C$      RT
006200      V$=D$      RT
006210 RETURN      RT
006220      RT
006230 DEFIN:      RT
006240 ACCEPT      RT
006250      AT (01,01),      :
006260 "CRES: Technical Word List Maintenance Subsystem",      :
006270      AT (01,51), FAC(HEX(8C)),      : JM
006280 EXP$,      :
006290      AT (02,01),"=====      :
006300 =====",      :
006310      AT (05,08), FAC(HEX(AD)),      ITLE5$      , CH(14),      JM
006320      AT (05,49), FAC(HEX(AD)),      ITLE6$      , CH(10),      JM
006330      AT (07,03), FAC(HEX(84)),      P$(1)      , CH(03),      :
006340      AT (07,08),      :
006350 "Adjective .....Modifies a noun or pronoun",      :
006360      AT (08,03), FAC(HEX(84)),      P$(2)      , CH(03),      :
006370      AT (08,08),      :
006380 "Adjective/verb .....A word that may function as an adjective"      :
006390 or verb",      :
006400      AT (09,03), FAC(HEX(84)),      P$(3)      , CH(03),      :

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 15:49 10/30/84 Page 12  
Input File is CRES0TWL in Library CRES0SRC on Volume CRES02

```

006410      AT (09,08),
006420 "Adverb .....Modifies a verb, adjective, or adverb",
006430      AT (10,03), FAC(HEX(84)), P$(4) , CH(03),
006440      AT (10,08),
006450 "Adverb/verb .....A word that may function as an adverb or
006460 verb",
006470      AT (11,03), FAC(HEX(84)), P$(5) , CH(03),
006480      AT (11,08),
006490 "Article .....A limiting adjective before a noun (e.g.
006500 a, the)",
006510      AT (12,03), FAC(HEX(84)), P$(6) , CH(03),
006520      AT (12,08),
006530 "Conjunction .....Links phrases or clauses (e.g. or)",
006540      AT (13,03), FAC(HEX(84)), P$(7) , CH(03),
006550      AT (13,08),
006560 "Interjection .....Used for simple exclamations (e.g. Oh!)"
006570 ,
006580      AT (14,03), FAC(HEX(84)), P$(8) , CH(03),
006590      AT (14,08),
006600 "Noun .....Indicates a person, place, or thing",
006610      AT (15,03), FAC(HEX(84)), P$(13) , CH(03),
006620      AT (15,08),
006630 "Noun/adjective .....A word that may function as a noun or ad
006640 jective",
006650      AT (16,03), FAC(HEX(84)), P$(9) , CH(03),
006660      AT (16,08),
006670 "Noun/verb .....Word that may function as a noun or verb"
006680 ",
006690      AT (17,03), FAC(HEX(84)), P$(10) , CH(03),
006700      AT (17,08),
006710 "Preposition .....Expresses a relationship between two ele
006720 ments of ",
006730      AT (18,08),
006740 " a sentence (e.g. in, on, by, to, since) JM
006750 ",
006760      AT (19,03), FAC(HEX(84)), P$(11) , CH(03),
006770      AT (19,08),
006780 "Pronoun .....Word that may be used as a substitute fo
006790 r a noun",
006800      AT (20,08),
006810 " (e.g. I, you, she, what,....)", JM
006820      AT (21,03), FAC(HEX(84)), P$(12) , CH(03),
006830      AT (21,08),
006840 "Verb .....Denotes an action or occurrence",
006850      AT (23,20),
006860 "Press",
006870      AT (23,26), FAC(HEX(84)), PF1$ , CH(03),
006880      AT (23,30),
006890 "to return to the previous screen",
006900 KEYS(BIN(1)&BIN(17)), ON (BIN(1)&BIN(17)) GOTO AOK,AOK
006910 EXIT: CLOSE #1 RT
006920 OUT: /* RETURNING TO TECHNICAL WORD LIST MENU */ JM

```

Print Technical Word List (CRES@TWP)

CRES@TWP in library CRES@SRC is the BASIC language program to print the technical word lists. The object language version is CRES@TWP in library CRES@SAV.

This program first allows the user to specify the technical word list to be printed. Then, the program enables the user to print out either the entire technical word list, or a specified portion. This printout can be displayed at the terminal or sent to the printer. An optional subtitle of up to two lines can be included for the printout, and the number of lines per page may be specified as some value less than the default of 50.

A complete program listing follows. Appendix C contains a sample flowchart.

Technical Report 86-001

THIS PAGE INTENTIONALLY LEFT BLANK.

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 11:37 11/09/84 Page 1  
Input File is CRES@IWP in Library CRES@SRC on Volume CRES02

```

000010 /* This is the CRES TECHNICAL WORD LIST PRINT SUBSYSTEM PROGRAM */ JM
000035 /* */ RT
000060 /* AUTHORS: Rose Tank and John Mears */ RT
000085 /* Date: October 12, 1983 */ RT
000110 /* */ RT
000135 RT
000160 /* The following code sets up a screen, which asks for the */ RT
000185 /* name of the file containing the technical word list */ RT
000210 /* to be printed. It also provides the opportunity to change */ RT
000235 /* the library and volume names which are set to the appro- */ RT
000260 /* priate defaults through the "Extract" system routine. */ RT
000285 RT
000310 RT
000335 DIM WORD$24, CODE$1, MSG$50, F$40, G$1, CMSG$255, MSG2$40 RT
000360 DIM IOBUFF$64, EXP$31, PFILE$08, RVOL$06, PF1$3, RLIB$8 JM
000385 DIM TAB$3, BTAB$7, FILE$08, LIB$08, FIL$19, SVOL$6, SLIB$8 JM
000410 DIM IVOL$06, HEADER$51, L$1, V$1, B$1, C$2, D$8, M$1, MSK$2 RT
000435 DIM SBTITLE$20, A$(14)10, SBTITL1$20, E2$41 JM
000460 DIM ENTER$ 05 RT
000485 DIM PF16$ , 04 RT
000510 DIM ER$ 79, MESSG$21 RT
000535 DIM STARTPH$ 20 RT
000560 DIM ENDPH$ 20 RT
000585 DIM TP$ 08 RT
000610 DIM PF15$ 04 RT
000660 A$(1)= "adjective" RT
000685 A$(2)= "verb" RT
000710 A$(3)= "conjugate" RT
000735 A$(4)= "adjec/verb" RT
000760 A$(5)= "noun" RT
000785 A$(6)= "pronoun" RT
000810 A$(7)= "interject" RT
000835 A$(8)= "noun/verb" RT
000860 A$(9)= "adverb" RT
000885 A$(10)= "preposit" RT
000910 A$(11)= "article" RT
000935 A$(12)= "adv/verb" RT
000960 A$(13)= "noun/adj" RT
000985 TAB$= "TAB" RT
001010 BTAB$= "BACKTAB" RT
001035 EXP$= "(Explanation of Entries Screen)" RT
001060 MESSG$= "Parts of Speech Codes" RT
001065 ENTER$= "ENTER" RT
001085 F$= "Filename not found on (library/volume)." RT
001090 E2$= " Please specify a TECHNICAL word list." JM
001110 CALL "EXTRACT" ADDR("IV", IVOL$, "RV", RVOL$, "IL", LIB$, "RL", RLIB$, : JM
001135 "SL", SLIB$, "SV", SVOL$) JM
001160 AGAIN: FILE$= " ": G$= HEX(81) RT
001185 MSG$= " ": L$= HEX(81) RT
001210 V$= HEX(81): M$= HEX(8D) RT
001235 HEADER$= "CRES TECHNICAL WORD LIST PRINT SUBSYSTEM" RT
001260 FIL$= "Technical Word List" JM
001310 PF1$= "PF1": PF16$= "PF16" RT
001335 RT
001360 /* Print the first screen at the terminal. */ RT
001385 RT
001410 FILMNU: RT

```

```

001435 ACCEPT
001460 AT (01,01),
001485 "CRES: Technical Word List Print Subsystem",
001510 AT (01,57),
001535 "(File Selection Screen)",
001560 AT (02,01),
001585 "=====
001610 =====",
001635 AT (04,12),
001660 "Welcome to the",
001685 AT (04,28), FAC(HEX(85)),
001710 HEADER$,
001735 AT (07,12),
001760 "Please specify the",
001785 AT (07,31), FAC(HEX(84)),
001810 FIL$,
001835 AT (07,51),
001860 "you want to print.",
002085 AT (13,30),
002110 "File =",
002135 AT (13,43),FAC(G$),FILE$, , CH(08),
002160 RANGE ( ":", " ", "cccccccccc"),
002185 AT (14,30),
002210 "Library =",
002235 AT (14,43),FAC(L$), LIB$, , CH(08),
002260 RANGE ( ":", " ", "cccccccccc"),
002285 AT (15,30),
002310 "Volume =",
002335 AT (15,43),FAC(V$), IVOL$, , CH(06),
002360 RANGE ( ":", " ", "cccccccccc"),
002385 AT (18,24), FAC(M$), MSG$,
002387 AT (20,30),"Press",
002389 AT (20,36),FAC(HEX(84)),
002391 ENTER$,
002393 AT(20,42),
002395 "to continue",
002397 AT (21,15),"Press",
002399 AT (21,21), FAC(HEX(84)),
002401 PFI6$,
002403 AT (21,26),
002405 "to return to the Technical Word List Menu",
002410 ON (BIN(16)&BIN(32)) GOTO OUT,OUT
002435
002460 /* SET UP THE USER FILE BLOCK */
002485
002510 SELECT #1,"TECHFI",VARC,INDEXED,RECSIZE=40,KEYPOS=1,KEYLEN=20,
002535 IOERR GOTO IOERR1
002560
002585 SELECT #2,"PRIFILE",PRINTER,RECSIZE=80,IOERR GOTO IOERR1
002610
002635 /* SEE IF THE SPECIFIED FILE EXISTS */
002660
002685 CALL "LIMITS"(FILE$,LIB$,IVOL$,CODE$,MSG$)
002710 IF CODE$="Y" THEN ZZZ
002735 IF CODE$="N" THEN GOSUB'7(F$,HEX(91),HEX(81),HEX(81))
002760 IF CODE$="V" THEN GOSUB'7(STR(MSG$,3),HEX(81),HEX(81),HEX(91))
002785 IF CODE$="L" THEN GOSUB'7(STR(MSG$,3),HEX(81),HEX(81),HEX(81))

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 11:37 11/09/84 Page 3  
Input File is CRES@TWP in Library CRES@SRC on Volume CRES02

```

002810 IF CODE$="E" THEN GOSUB'7(STR(MSG$,3),HEX(81),HEX(81),HEX(81)) RT
002835 GOTO FILMNU RT
002860 RT
002885 /* IF WE FALL HERE, IT MUST BE TIME TO OPEN FILE */ RT
002910 RT
002915 ZZZ: CALL "READFDR" ADDR(FILE$,LIB$,IVOL$,0%,"RS",RAG%,RET%) JM
002925 IF RAG%=40 THEN ZZ1 JM
002927 GOSUB'7(E2$,HEX(81),HEX(81),HEX(81)) JM
002929 GOTO FILMNU JM
002935 ZZ1: OPEN NOGETPARM #1,SHARED, FILE= FILE$, LIBRARY = LIB$, ! JM
002960 VOLUME = IVOL$ RT
002985 RT
003010 READ #1,IOBUFF$, DATA GOTO IOERR2 RT
003035 IF STR(IOBUFF$,1,20)= " TECHNICAL WORDS " THEN YES JM
003060 CLOSE #1 RT
003085 M$=HEX(94) RT
003110 MSG$= " Please specify a TECHNICAL word list." RT
003135 GOTO FILMNU RT
003160 YES: RT
003185 /* This section of code generates the second screen, which asks */ RT
003210 /* for the particular word interval and its corresponding */ RT
003235 /* print operation. The user is given the option of returning */ RT
003260 /* to the previous screen to select a new word file to print, */ RT
003285 /* or of returning to the technical word list menu. */ RT
003310 RT
003335 _R$ = " " RT
003360 RT
003385 CWMENU: RT
003410 WORD$ = " " RT
003435 LPP% =50 RT
003460 ENTER$ = "ENTER" RT
003485 PF15$ = "PF15" RT
003510 PF16$ = "PF16" RT
003535 SBTITLE$ = " " RT
003560 SBTITL1$ = " " RT
003585 BALONEY: RT
003610 STARTPH$ = "FIRST" RT
003635 ENDPH$ = "LAST" RT
003660 TP$ = "TERMINAL" RT
003685 PRINTSCR: RT
003710 ACCEPT : RT
003735 AT (01,03), : RT
003760 "CRES: Technical Word List Print Subsystem", : RT
003785 AT (01,66), : RT
003810 "(Print Screen)", : RT
003835 AT (02,03), : RT
003860 "===== : RT
003885 =====, : RT
003910 AT (04,13), : RT
003935 "To print a range of lines, fill in the information below.", : RT
003960 AT (06,02), FAC(HEX(85)), : RT
003985 ER$, : RT
004010 AT (08,22), : RT
004035 "Subtitle of list =", : RT
004060 AT (08,42), FAC(HEX(81)),SBTITLE$,CH(20), : RT
004085 AT (09,42), FAC(HEX(81)),SBTITL1$,CH(20), : RT
004110 AT (12,22), : RT

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 11:37 11/09/84 Page 4  
Input File is CRES@WIP in Library CRES@SRC on Volume CRES02

```

004135 "Starting word" =",
004160 AT (12,42), FAC(HEX(81)), STARTPH$, CH(20),
004185 RANGE ( "!" " , "XXXXXXXXXX"),
004210 AT (13,22),
004235 "Ending word" =",
004260 AT (13,42), FAC(HEX(81)), ENDPH$, CH(20),
004285 RANGE ( "!" " , "XXXXXXXXXX"),
004310 AT (14,22),
004335 "Lines per page" =",
004360 AT (14,42), FAC(HEX(82)), LPP$, PIC(##),
004385 AT (17,22),
004410 "Print at (TERMINAL/PRINTER) =",
004435 AT (17,53), FAC(HEX(81)), TP$, CH(08),
004460 AT (22,32),
004485 "Press",
004510 AT (22,38), FAC(HEX(85)), ENTER$ , CH(05),
004535 AT (22,44),
004560 "to print",
004585 AT (23,18),
004610 "Press",
004635 AT (23,24), FAC(HEX(84)), PF15$, CH(04),
004660 AT (23,29),
004685 "to select another Technical Word List",
004710 AT (24,18),
004735 "Press",
004760 AT (24,24), FAC(HEX(84)), PF16$, CH(04),
004785 AT (24,29),
004810 "to return to Technical Word List Menu",
004835 KEYS(BIN(0)&BIN(15)&BIN(16)&BIN(31)&BIN(32)),
004860 ON (BIN(15)&BIN(31)&BIN(16)&BIN(32)) GOTO NEW,NEW,EXIT,EXIT
004885
004910 /* Make sure parameters input are correct */
004935
004960 IF STR(TP$,1,1) = "P" THEN NEXT11
004985 IF STR(TP$,1,1) <> "T" THEN TERMPR
005010 NEXT11: IF LPP% < 51 THEN NEXT13
005035 ER$="Lines per page can be at most 50."
005060 GOTO PRINTSCR
005085
005110 IOERR1: PRINT /* This loop used if I/O error */
005135 PRINT BELL,HEX(94)
005160 CLOSE #1
005185 FOR I=1 TO 200000 :NEXT I
005210 GOTO FILEMENU
005235
005260 TERMPR:
005285 ER$="Print at ' has only legal values TERMINAL or PRINTER."
005310 GOTO PRINTSCR
005335
005360 REX: ER$=" 'Starting word' comes after 'Ending word'."
005385 GOTO PRINTSCR
005410
005435
005460 IOERR2: PRINT
005485 PRINT HEX(94),"error occurred on file "; FILE$
005510 PRINT HEX(94),"Error code = "; FS(#1)
005535 STOP "Touch ENTER to return to menu."

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 11:37 11/09/84 Page 5  
Input File is CRES@TWP in Library CRES@SRC on Volume CRES02

```

005560          GOTO CWMENU                                RT
005585                                                    RT
005610 NEXT13: IF STARTPH$="FIRST" OR ENDPH$="LAST" THEN FORD RT
005635          IF STARTPH$>ENDPH$ THEN REX                  RT
005660                                                    RT
005685 /* Now it's time to generate our print file name */ RT
005710                                                    RT
005735 FORD:   DAT$= DATE                                    RT
005760          D$= STR(DAT$,3,2)&"/"&STR(DAT$,5,2)&"/"&STR(DAT$,1,2) RT
005785          T$= STR(TIME,1,2)&":"&STR(TIME,3,2)           RT
005810                                                    RT
005835 DISPLAY                                         : JM
005860          AT (01,03),                                   :
005885 "CRES: Technical Word List Print Subsystem",       : JM
005910          AT (01,61),                                   :
005935 "(Generation Screen)",                             : JM
005960          AT (02,03),                                   :
005985 "=====":                                         :
006010 "=====":                                         :
006035          AT (12,23),                                   :
006060 "Output file now being generated!"                 JM
006085          PFILE$="TECHWD##"                             RT
006110          I%=0                                           RT
006135 NEXT14: CONVERT I% TO STR(PFILE$,7,2),PIC(00)       RT
006160          I%=I%+1                                       RT
006210          CALL "LIMITS" (PFILE$,SLIB$,SVOL$,CODE$,MSG$) JM
006235          IF CODE$="Y" THEN NEXT14                       RT
006245          IF CODE$="E" THEN ER$ = STR(MSG$,3)           JM
006255          IF CODE$="E" THEN PRINTSCR                   JM
006260          OPEN NOGETPARM #2,OUTPUT,FILE=PFILE$,LIBRARY=SLIB$, : JM
006285          VOLUME=SVOL$,SPACE=12000                     JM
006310                                                    JM
006335          GOSUB TTL PAGE                                JM
006360          IF STARTPH$="FIRST" THEN READ #1,KEY>" TECHNICAL WORDS", : JM
006385          IOBUFF$ ELSE READ #1,KEY>STARTPH$, IOBUFF$    JM
006410          IF ENDPH$="LAST" THEN ENDPH$=ALL(HEX(FF))    JM
006435          PAGE% = 1                                       JM
006460          RECS% = 1                                       JM
006485          GOSUB HEADER                                  JM
006510          IF STR(IOBUFF$,1,20)>ENDPH$ THEN HOME        JM
006535          GOTO KNOT                                       JM
006560 AGAN:   READ #1, IOBUFF$,EOD GOTO HOME               JM
006585          IF STR(IOBUFF$,1,20)>ENDPH$ THEN HOME        JM
006610          IF RECS% = 1 AND PAGE% <> 1 THEN GOSUB HEADER  JM
006635 KNOT:   IF STR(IOBUFF$,1,20)=" TECHNICAL WORDS" THEN AGAN JM
006660          PSN= POS("JVCJNOInAppad"= STR(IOBUFF$,21,1)) JM
006665          IF PSN=0 THEN PSN=14                           JM
006667          IF PSN=14 THEN A$(14)=STR(IOBUFF$,21,1)       JM
006685          IF STR(IOBUFF$,22,1)="*" THEN STR(IOBUFF$,22,1)="P" JM
006710                                                    JM
006735 /* Do we need '+=Root Word' or 'BOLDFACE=Root Word'? */
006760                                                    RT
006785          IF STR(TP$,1,1)="T" THEN BUG                    :
006810          WRITE #2, USING CAT6,C$," ",STR(IOBUFF$,1,20), : RT
006835          A$(PSN),STR(IOBUFF$,22,1),STR(IOBUFF$,27,1)    RT
006860          IF (STR(IOBUFF$,26,1)<>"*") THEN BAG
006885

```

AD-A168 681

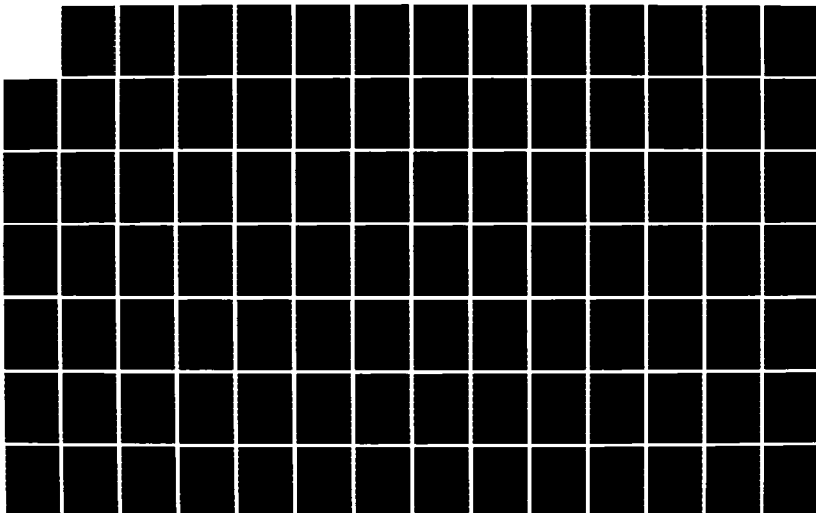
COMPUTER READABILITY AND EDITING SYSTEM (CRES)  
PROGRAMMER'S REFERENCE MANUAL(U) NAVAL TRAINING SYSTEMS  
CENTER ORLANDO FL S R SCHALOW ET AL. JAN 86  
NAVTRASYSCEN-TR-86-001

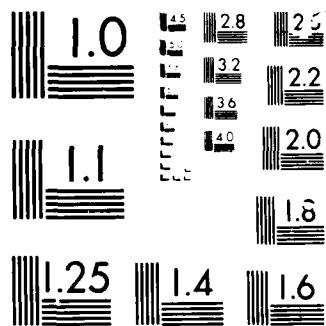
2/4

UNCLASSIFIED

F/G 5/9

NL





MICROCOPY

CHART

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 11:37 11/09/84 Page 6  
Input File is CRES@TWP in Library CRES@SRC on Volume CRES02

```

006910 /* If this is a rootword at printer, make word BOLDFACE */
006935
006960 C$=HEX(0000)
006985 WRITE #2, USING CAT9, C$, STR(IOBUFF$,1,20)
007010 WRITE #2, USING CAT9, C$, STR(IOBUFF$,1,20)
007035 C$=HEX(0001)
007060 GOTO BAG
007085
007110 /* Turn ROOTWORD's "*" into "+" for TERMINAL output! */
007135
007160 AUG: IF STR(IOBUFF$,26,1)="" THEN STR(IOBUFF$,26,1)="+"
007185 WRITE #2, USING CAT6,C$,STR(IOBUFF$,26,1),STR(IOBUFF$,1,20),: RT
007210 AS(PSN),STR(IOBUFF$,22,1),STR(IOBUFF$,27,1) RT
007235 RT
007260 BAG: RECS%=RECS%+1 RT
007285 IF RECS%>LPP% THEN RECS%=1 RT
007310 IF RECS%=1 THEN PAGE%=PAGE%+1 RT
007335 GOTO AGAN RT
007360
007385 HOME: /*,Need to print 'END OF LISTING' and '(?)=Root Word' */
007410
007435 IF RECS%=1 THEN RECS%=51 RT
007460 WRITE #2, USING DOG, C$ RT
007485 RECS%=RECS%+2 RT
007510 WRITE #2, USING DG, C$ RT
007535 FOR I = RECS% TO LPP%+1 RT
007560 WRITE #2, USING DOG, C$ RT
007585 NEXT I RT
007610 IF STR(TP$,1,1)="P" THEN HER1
007635 WRITE #2, USING CAT8, C$ RT
007660 GOTO HER
007685 HER1: WRITE #2, USING RAT, C$
007710 C$=HEX(0000)
007735 FOR I=1 TO 4
007760 WRITE #2, USING RAT1, C$
007785 NEXT I
007810 C$ = HEX(0001)
007835 HER: CLOSE #2 RT
007860 IF STR(TP$,1,1)="P" THEN MESS RT
007885 CLOSE WS RT
007910 MSG2$ = "CANCEL Processing" RT
007935 MSK$=HEX(00) RT
007960 CMSG$ = " " RT
007985 CALL "LINK" ADDR("DISPFILE","P",RLIB$,RVOL$,3%,PFILE$, : JM
008010 SLIB$,SVOL$, "C",MSG2$,27%, "H",MSK$,CMSG$,255%,C$,R%) JM
008035 JM
008060 ER$ = " " RT
008085 IF C%<>0 OR R%<>0 : RT
008110 THEN ER$ = STR(CMSG$,141) RT
008135 GOTO CMENU RT
008210 JM
008235 MESS: ER$=" File "&PFILE$& " in Library "&SLIB$& " on "& : JM
008260 SVOL$& " generated." JM
008285 GOTO BALONEY RT
008310 RT
008335 NEW: PRINT PAGE, AT(12,28),HEX(84);"Closing last file" RT
008360 FOR I= 1 TO 50000 :NEXT I RT

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 11:37 11/09/84 Page 7  
Input File is CRES@TWP in Library CRES@SRC on Volume CRES02

```

008385      CLOSE #1                                RT
008410      GOTO AGAIN                              RT
008435                                           RT
008460 TTL_PGE:                                RT
008485      C$=HEX(8001)                            RT
008510      WRITE #2, USING DOG, C$                 RT
008535      C$ = HEX(0015)                          RT
008560      WRITE #2, USING DOG, C$                 RT
008585      C$ = HEX(0001)                          RT
008610      WRITE #2, USING DOG1, C$                RT
008635      WRITE #2, USING DOG2, C$                RT
008660      WRITE #2, USING DOG0, C$,D$,T$          RT
008685      WRITE #2, USING DOG3, C$                RT
008710      WRITE #2, USING DOG2, C$                RT
008735      WRITE #2, USING DOG1, C$                RT
008760      WRITE #2, USING DOG2, C$                RT
008785      IF SBTITLE$ = " " THEN GERRY           RT
008810      WRITE #2, USING DOG4, C$, SBTITLE$      RT
008835      IF SBTITL1$<>" " THEN WRITE #2, USING : RT
008860 DOG10, C$, SBTITL1$                        RT
008885      WRITE #2, USING DOG2, C$                RT
008910 GERRY: WRITE #2, USING DOG5, C$, FILE$      RT
008935      WRITE #2, USING DOG2, C$                RT
008960      WRITE #2, USING DOG6, C$, LIB$          RT
008985      WRITE #2, USING DOG2, C$                RT
009010      WRITE #2, USING DOG7, C$, IVOL$         RT
009035      WRITE #2, USING DOG2, C$                RT
009060      WRITE #2, USING DOG8, C$, STARTPH$      RT
009085      WRITE #2, USING DOG2, C$                RT
009110      WRITE #2, USING DOG9, C$, ENDPH$        RT
009135      WRITE #2, USING DOG2, C$                RT
009160      WRITE #2, USING DOG1, C$                RT
009185      RETURN                                  RT
009210                                           RT
009235 HEADER:                                RT
009260      IF PAGE% = 1 THEN FIRST                 RT
009285      C$ = HEX(0001)                          RT
009310      IF STR(TP$,1,1) = "P" THEN FRST1       RT
009335      WRITE #2, USING DOG, C$                 RT
009360      WRITE #2, USING CAT8, C$                 RT
009385      GOTO FIRST                              RT
009410 FRST1: WRITE #2, USING DOG, C$              RT
009435      WRITE #2, USING RAT, C$                 RT
009460      C$=HEX(0000)                            RT
009485      FOR I = 1 TO 2                          RT
009510      WRITE #2, USING RAT1, C$                RT
009535      NEXT I                                    RT
009560 FIRST: C$ = HEX(8001)                        RT
009585      WRITE #2, USING DOG, C$                 RT
009610      C$ = HEX(0001)                          RT
009635      IF SBTITLE$ = " " THEN WRITE #2, USING CAT,C$,D$,T$,PAGE% : RT
009660      ELSE FOUT                                : RT
009685      GOTO ALRITE                               RT
009710      WRITE #2, USING CAT1,C$,SBTITLE$,D$,T$,PAGE% RT
009735      IF SBTITL1$<>" " THEN WRITE #2, USING CAT7,C$,SBTITL1$ : RT
009760      THEN WRITE #2, USING CAT7,C$,SBTITL1$    RT
009785

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 11:37 11/09/84 Page 8  
Input File is CRES@TWP in Library CRES@SRC on Volume CRES02

```

009810 ALRITE: WRITE #2, USING CAT2, C$, FILE$, LIB$, IVOL$ RT
009835 WRITE #2, USING CAT3, C$ RT
009860 WRITE #2, USING CAT4, C$ RT
009885 WRITE #2, USING CAT5, C$ RT
009910 WRITE #2, USING CATY, C$ RT
009935 RETURN RT
009960 RT
009985 DOG0:### : ##### ! RT
010010 ##### : RT
010035 DOG: ### RT
010060 DOG1:### : ----- ! RT
010085 ----- ! RT
010110 DOG2:### : ! RT
010135 : ! RT
010160 DOG3:### : ***** TECHNICAL WORD LIST *** ! RT
010185 ** : RT
010210 DOG4:### : TITLE OF LIST: ##### ! RT
010235 ##### : RT
010260 DOG5:### : INPUT FILE: ##### ! RT
010285 : RT
010310 DOG6:### : INPUT LIBRARY: ##### ! RT
010335 : RT
010360 DOG7:### : INPUT VOLUME: ##### ! RT
010385 : RT
010410 DOG8:### : STARTING WORD: ##### ! RT
010435 : RT
010460 DOG9:### : ENDING WORD: ##### ! RT
010485 ##### : RT
010510 DOG10:### : ##### ! RT
010535 ##### : RT
010560 DOG:### !!! END OF LISTING !!! RT
010585 CAT:### ##### PAGE ##### RT
010610 CAT1:### TITLE OF LIST : ##### ! RT
010635 ##### PAGE ##### RT
010660 CAT7:### ##### RT
010685 CAT2:### Input File is ##### in Library ##### on Volume ### RT
010710 ### RT
010735 CAT3:###===== ! RT
010760 ===== RT
010785 CAT4:### WORD Part of Past ! RT
010810 Mandatory RT
010835 CATY:###===== ! RT
010860 ===== RT
010885 CAT5:### Speech Participle ! RT
010910 Technical word RT
010935 CAT6:### # ##### # ! RT
010960 # RT
010985 RT
011010 CAT9:### ##### RT
011035 CAT8:### + = Root Word RT
011060 RAT:### BOLDFACE = Root Word
011085 RAT1:### BOLDFACE
011110 RT
011135 /* THIS FUNCTION PRINTS OUT ERROR MESSAGES AND */ RT
011160 /* SETS UP FLASHING FAC'S IN APPROPRIATE PLACES */ RT
011185 DEF FN"(MGS$,B$,C$,D$) RT
011210 PRINT RT

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 11:37 11/09/84 Page 9  
Input File is CRES@IWP in Library CRES@SRC on Volume CRES02

011235	MS=HEX(94)	RT
011260	CS=B\$	RT
011285	LS=C\$	RT
011310	VS=D\$	RT
011335	RETURN	RT
011360		RT
011385	EXIT: CLOSE #1	RT
011410	OUT: /* RETURNING TO TECHNICAL WORD LIST MENU */	RT

Technical Report 86-001

THIS PAGE INTENTIONALLY LEFT BLANK.

Print Technical Words By Attribute (CRES@TPA)

CRES@TPA in library CRES@SRC is the BASIC language program to print the technical word list by attribute. The object language version is CRES@TPA in library CRES@SAV.

This program first allows the user to specify the technical word list to be printed and the attributes, (part of speech, past participle, form of the verb "to be", negative or root word), to be included. See table 1 for a definition of the parts of speech.

If a particular part of speech is selected which also appears in combination with some other part(s) of speech, all combinations which contain the originally specified part of speech will be included. (For example, selecting verb will produce a list of verbs, adverb/verbs, noun/verbs, and adjective/verbs.) More than one attribute can be selected, and all words containing any attribute selected will be included in the printout. If the root word attribute is selected in combination with another attribute, then only root words possessing that attribute will be printed.

The user can select a subtitle of up to two lines to be printed on each page of the output listing. The output listing can be sent to either the terminal or the printer, it can encompass the entire output list generated, or any portion thereof, and the number of lines per page can be specified.

A complete program listing follows. Appendix C contains a sample flowchart.

Technical Report 86-001

THIS PAGE INTENTIONALLY LEFT BLANK.

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 14:56 11/13/84 Page 1  
Input File is CRES@TPA in Library CRES@SRC on Volume CRES02

```

000010 /* This is the CRES TECHNICAL PRINT BY ATTRIBUTE PROGRAM */ JM
000030 /* */
000050 /* AUTHORS: Rose Tank and John Mears */
000070 /* Date: October 12, 1983 */
000090 /* */
000110
000130 /* The following code sets up a screen, which asks for the */
000150 /* name of the file containing the Technical word List */ JM
000170 /* to be printed. It also provides the opportunity to change */
000190 /* the library and volume names which are set to the appro- */
000210 /* priate defaults through the "Extract" system routine. */
000230
000250
000270 DIM WORD$24,ERR1$60,CODE$1,MSG$50,F$40,G$1,CM$255,MSG2$40
000290 DIM F1$20,IOBUFF$64,RVOL$06,PFILE$08,FILE$8,LIB$8,IVOL$6 JM
000310 DIM ENTER$5,PF14$4,PF15$4,PF16$4,ER$79,FIL$19,E2$38 JM
000330 DIM HEADER$51,L$1,V$1,B$1,C$2,D$8,M$1,MSK$2
000350 DIM SBTITLE$20,AS(20)15,SBTITL1$20,STARIPH$20,ENDPH$20,TP$8
000370 DIM P$PCFIT$15,OTHERS$16,TS(20)1,RLIB$8,SLIB$8,SVOL$6 JM
000390
000410 AS(1)="adjective"
000430 AS(2)="verb"
000450 AS(3)="conjunct."
000470 AS(4)="adj/verb"
000490 AS(5)="noun"
000510 AS(6)="pronoun"
000530 AS(7)="interject"
000550 AS(8)="noun/verb"
000570 AS(9)="adverb"
000590 AS(10)="preposit"
000610 AS(11)="article"
000630 AS(12)="adv/verb"
000650 AS(13)="noun/adj" RT
000670 AS(14)="past participle" RT
000690 AS(15)="to be" RT
000710 AS(16)="negative" RT
000730 AS(17)="mand.tech.word" RT
000750 AS(18)="root word" RT
000770
000790 F1$=HEX(00140001000100010003A00100010003000B)
000810 ENTER$="ENTER": PF14$="PF14":PF15$="PF15":PF16$="PF16" RT
000812 ATTRIBUTES$="ATTRIBUTES" RT
000830 P$PCFIT$="Parts of Speech"
000850 OTHERS$="Other Attributes"
000870 F$="Filename not found on (library/volume)."
000890 E2$=" Please specify a TECHNICAL word list." JM
000900 ERR1$=" I/O ERROR - Please specify a TECHNICAL word list." JM
000910 HEADER$="CRES PRINT FWL BY ATTRIBUTE SUBSYSTEM" JM
000930 FIL$="Technical word List"
000950 CALL "EXTRACT" ADDR("RV",RVOL$,"RL",RLIB$,"SL",SLIB$,"SV",SVOL$, ! JM
000952 "IL",LIB$,"IV",IVOL$) JM
000970 FILE$=" "
001010 AGAIN: G$=HEX(81)
001030 MSG$=" ":L$=HEX(81)
001050 V$=HEX(81):M$=HEX(8D)
001070
001090 FIL4NU:

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 14:56 11/13/84 Page 2  
Input File is CRES@TPA in Library CRES@SRC on Volume CRES02

```

001110 ACCEPT
001130 AT (01,01),
001150 "CRES: Print Technical Word List by Attribute Subsystem",
001170 AT (01,57),
001190 "(File Selection Screen)",
001210 AT (02,01),
001230 "=====
001250 =====",
001270 AT (04,19),
001290 "Welcome to the",
001310 AT (04,30), FAC(HEX(85)),
001330 HEADERS$,
001350 AT (07,12),
001370 "Please specify the",
001390 AT (07,31), FAC(HEX(84)),
001410 FILE$,
001430 AT (07,51),
001450 "you want to print.",
001510 AT (13,30),
001530 "File =",
001550 AT (13,43),FAC(G$),FILE$, CH(08),
001570 RANGE ( ":", " ", "XXXXXXXXXX"),
001590 AT (14,30),
001610 "Library =",
001630 AT (14,43),FAC(L$), LIB$, CH(08),
001650 RANGE ( ":", " ", "XXXXXXXXXX"),
001670 AT (15,30),
001690 "Volume =",
001710 AT (15,43),FAC(V$), IVOL$, CH(06),
001730 RANGE ( ":", " ", "XXXXXXXXXX"),
001750 AT (18,24), FAC(M$), MSG$,
001770 AT (20,30),"Press",
001790 AT (20,36),FAC(HEX(84)),
001810 ENTER$,
001830 AT(20,42),
001850 "to continue",
001870 AT (21,15),"Press",
001890 AT (21,21), FAC(HEX(84)),
001910 PF16$,
001930 AT (21,26),
001950 "to return to the Technical Word List Menu",
001970 ON (BIN(16)&BIN(32)) GOTO OUT
002010
002030 M$=HEX(80):MSG$=" "
002050 /* SET UP THE USER FILE BLOCK */
002070
002090 SELECT #1,"TECHNIC",VARC,INDEXED,RECSIZE=40,KEYPOS=1,KEYLEN=20, :
002110 IOERR GOTO IOERR1
002130
002150 SELECT #2,"PRFILE",PRINTER,RECSIZE=80,IOERR GOTO IOERR1
002170
002190 /* SEE IF THE SPECIFIED FILE EXISTS */
002210
002230 CALL "LIMITS"(FILE$,LIB$,IVOL$,CODE$,MSG$)
002250 IF CODE$="Y" THEN ZZZ
002270 IF CODE$="N" THEN GOSUB 7,F$,HEX(91),HEX(81),HEX(81))
002290 IF CODE$="V" THEN GOSUB 7(STR(MSG$,3),HEX(81),HEX(81),HEX(91))

```

JM

RT

RT

RT

RT

RT

RT

RT

RT

RT

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 14:56 11/13/84 Page 3  
Input File is CRES@TPA in Library CRES@SRC on Volume CRES02

```

002310 IF CODE$="L" THEN GOSUB'7(STR(MSG$,3),HEX(81),HEX(91),HEX(81))
002330 IF CODE$="E" THEN GOSUB'7(STR(MSG$,3),HEX(81),HEX(81),HEX(81))
002350 GOTO FILMNU
002370
002390 /* OUR FILE EXISTS, BUT IS IT A TECHNICAL WORD LIST? */ JM
002414 JM
002416 ZZZ: CALL "READFDR" ADDR(FILE$,LIB$,IVOL$,0%,"RS",RAG$,RET%) JM
002418 IF RAG%=40 THEN ZZ1 JM
002420 GOSUB'7(E2$,HEX(81),HEX(81),HEX(81)) JM
002422 GOTO FILMNU JM
002430 ZZ1: OPEN NOGETPARM #1,SHARED, FILE= FILE$, LIBRARY = LIB$, : JM
002450 VOLUME = IVOL$
002490 READ #1,IOBUFF$, DATA GOTO IOERR2
002510 IF STR(IOBUFF$,16)= " TECHNICAL WORDS" THEN YES
002530 CLOSE #1
002550 M$=HEX(94)
002570 MSG$= E2$ JM
002590 GOTO FILMNU
002610 YES:
002630 X$="X": RTFL=0: FOR I=1 TO 18: T$(I)=" ":NEXT I RT
002650
002670 ACCEPT : RT
002690 AT (01,03), : RT
002710 "CRES: Print TWI, by Attribute Subsystem (Attribute Sele: JM
002730 ction Screen)", : RT
002750 AT (02,03), : RT
002770 "===== : RT
002790 =====", : RT
002810 AT (04,03), : RT
002830 " Place an", : RT
002850 AT (04,14),FAC(HEX(85)),X$, CH(01), : RT
002870 AT (04,16), : RT
002890 "beside the attribute or combination of attributes to be printed. : RT
002910 ", : RT
002930 AT (05,03), : RT
002950 "If you choose a part of speech which is included in a combinatio: RT
002970 n, then the", : RT
002990 AT (06,03), : RT
003010 "combination will also be printed. (e.g. If adverb is selected, t: RT
003030 hen adv/verbs", : RT
003050 AT (07,03), : RT
003070 "will be printed as well as the adverbs.) If the root word optio: RT
003090 n is selected ", : RT
003110 AT (08,03), : RT
003130 "in combination with another attribute, then only root words with: RT
003150 that ", : RT
003170 AT (09,03), : RT
003190 "attribute will be printed.", : RT
003210 AT (11,37), FAC(HEX(AD)), ATTRIBUTES$ , CH(10), : RT
003230 AT (13,14), FAC(HEX(AC)), PSPCTIT$ , CH(15), : RT
003250 AT (13,55), FAC(HEX(AC)), OTHERS$ , CH(16), : RT
003270 AT (14,05), : RT
003290 "adjective", : RT
003310 AT (14,18), FAC(HEX(81)), T$(1) , CH(01), : RT
003330 AT (14,24), : RT
003350 "noun/verb", : RT
003370 AT (14,36), FAC(HEX(81)), T$(8) , CH(01), : RT

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 14:56 11/13/84 Page 4  
Input File is CRES0TPA in Library CRES0SRC on Volume CRES02

003390	AT (14,55),			RT
003410	"past participle",			RT
003430	AT (14,71), FAC(HEX(81)),	TS(14)	, CH(01),	RT
003450	AT (15,05),			RT
003470	"verb",			RT
003490	AT (15,18), FAC(HEX(81)),	TS(2)	, CH(01),	RT
003510	AT (15,24),			RT
003530	"adverb",			RT
003550	AT (15,36), FAC(HEX(81)),	TS(9)	, CH(01),	RT
003570	AT (15,55),			RT
003590	"to be",			RT
003610	AT (15,71), FAC(HEX(81)),	TS(15)	, CH(01),	RT
003630	AT (16,05),			RT
003650	"conjunction",			RT
003670	AT (16,18), FAC(HEX(81)),	TS(3)	, CH(01),	RT
003690	AT (16,24),			RT
003710	"preposition",			RT
003730	AT (16,36), FAC(HEX(81)),	TS(10)	, CH(01),	RT
003750	AT (16,55),			RT
003770	"negative",			RT
003790	AT (16,71), FAC(HEX(81)),	TS(16)	, CH(01),	RT
003810	AT (17,05),			RT
003830	"adj/verb",			RT
003850	AT (17,18), FAC(HEX(81)),	TS(4)	, CH(01),	RT
003870	AT (17,24),			RT
003890	"article",			RT
003910	AT (17,36), FAC(HEX(81)),	TS(11)	, CH(01),	RT
003930	AT (17,55),			RT
003950	"mand tech words",			JM
003970	AT (17,71), FAC(HEX(81)),	TS(17)	, CH(01),	RT
003990	AT (18,05),			RT
004010	"noun",			RT
004030	AT (18,18), FAC(HEX(81)),	TS(5)	, CH(01),	RT
004050	AT (18,24),			RT
004070	"adv/verb",			RT
004090	AT (18,36), FAC(HEX(81)),	TS(12)	, CH(01),	RT
004110	AT (18,55),			RT
004130	"root word",			RT
004150	AT (18,71), FAC(HEX(81)),	TS(18)	, CH(01),	RT
004170	AT (19,05),			RT
004190	"pronoun",			RT
004210	AT (19,18), FAC(HEX(81)),	TS(6)	, CH(01),	RT
004230	AT (19,24),			RT
004250	"noun/adj",			RT
004270	AT (19,36), FAC(HEX(81)),	TS(13)	, CH(01),	RT
004290	AT (20,05),			RT
004310	"interjection",			RT
004330	AT (20,18), FAC(HEX(81)),	TS(7)	, CH(01),	RT
004350	AT (21,24), FAC(M\$),	MSG\$	, CH(45),	RT
004370	AT (22,30),			JM
004390	"press",			RT
004410	AT (22,36), FAC(HEX(85)),	ENTER\$	, CH(05),	JM
004430	AT (22,42),			JM
004450	"no continue",			RT
004470	AT (23,17),			JM
004490	"Press",			RT
004510	AT (23,23), FAC(HEX(84)),	PF15\$	, CH(04),	JM

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 14:56 11/13/84 Page 5  
Input File is CRES@TPA in Library CRES@SRC on Volume CRES02

```

004530      AT (23,28),                      : JM
004550 "to select another Technical Word List", : RT
004552      AT (24,15),                      : JM
004554 "Press",                             : JM
004556      AT (24,21), FAC(HEX(84)),         PF16$      , CH(04), : JM
004558      AT (24,26),                      : JM
004560 "to return to the Technical Word List Menu", : JM
004570 KEYS(BIN(0)&BIN(15)&BIN(31)&BIN(16)&BIN(32)), : JM
004590 ON (BIN(15)&BIN(31)&BIN(16)&BIN(32)) GOTO NEW,NEW,EXIT,EXIT : JM
004610
004630 MSG$=" ":MS=HEX(8D): K=0
004650 FOR I = 1 TO 17                      RT
004670   PSN=POS(" X"=T$(I))
004690   IF PSN=0 THEN INPTER
004710   IF PSN=2 THEN K=1
004730 NEXT I
004750 IF K=1 THEN 4852                      RT
004770 IF K=0 AND T$(18)="X" THEN RTPL=2 ELSE INPTER RT
004790 GOTO 4852                            RT
004810 INPTER: MSG$="Please use only 'X' to mark attributes."
004830 MS=HEX(94): GOTO YES
004850
004852 IF T$(1)<>"X" THEN 4856              /*If adj was marked then */ RT
004854   T$(4)="X": T$(13)="X"              /*also print adj/vero & noun/adj*/ RT
004856 IF T$(2)<>"X" THEN 4862              /*If vero was marked then also */ RT
004858   T$(4)="X": T$(8)="X": T$(12)="X"/*print adj/vero, noun/vero*/ RT
004860   /* & adv/vero */ RT
004862 IF T$(5)<>"X" THEN 4866              /*If noun was marked then also */ RT
004864   T$(8)="X": T$(13)="X"              /*print noun/vero & noun/adj */ RT
004866 IF T$(9)="X" THEN T$(12)="X"        /*If adv was marked then also */ RT
004868   /*print adv/vero */ RT
004870 RT
004890 FOR I =14 TO 17: IF T$(I)="X" THEN T$(I)="*" ELSE T$(I)="$":NEXT I RT
004910 IF K=1 AND T$(18)="X" THEN RTPL=1 RT
004930 /* This section of code generates the third screen, which asks */ JM
004950 /* for the particular word interval and its corresponding */ JM
004970 /* print operation. The user is given the option of returning */ JM
004990 /* to the previous screen to select different attributes, or of */ JM
004992 /* returning to the first screen to select a different word file*/ JM
005010 /* or of returning to the Technical Word List Menu. */ JM
005030
005050
005070 ER$ = " "
005090 CMENU:
005110 WORD$ = " "
005130 LPP$ =50
005150 PF15$ = "PF15"
005170 PF16$ = "PF16"
005190 SBTITLE$ = " "
005210 SBTITLE$ = " "
005230 BALONEY:
005250   FP$ = "TERMINAL"
005270   STARTPH$ = "FIRST"
005290   ENDPH$ = "LAST"
005310 PRINTSCR:
005330 ACCEPT
005350      AT (01,03),

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 14:56 11/13/84 Page 6  
Input File is CRES@TPA in Library CRES@SRC on Volume CRES02

```

005370 "CRES: Print Technical Word List by Attribute Subsystem",      JM
005390     AT (01,66),
005410 "(Print Screen)",
005430     AT (02,03),
005450 "=====
005470 =====",
005490     AT (04,13),
005510 "To print a range of lines, fill in the information below.",
005530     AT (06,02), FAC(HEX(85)),
005550 ER$,
005570     AT (08,22),
005590 "Subtitle of list =",
005610     AT (08,42), FAC(HEX(81)), S3TITLE$, CH(20),
005630     AT (09,42), FAC(HEX(81)), S3TITL1$, CH(20),
005650     AT (12,22),
005670 "Starting word =",
005690     AT (12,42), FAC(HEX(81)), STARTPH$, CH(20),
005710     RANGE ( ":", " ", "XXXXXXXXXX"),
005730     AT (13,22),
005750 "Ending word =",
005770     AT (13,42), FAC(HEX(81)), ENDPH$, CH(20),
005790     RANGE ( ":", " ", "XXXXXXXXXX"),
005810     AT (14,22),
005830 "Lines per page =",
005850     AT (14,42), FAC(HEX(82)), LPP$, PIC(##),
005870     AT (17,22),
005890 "Print at (TERMINAL/PRINTER) =",
005910     AT (17,53), FAC(HEX(81)), TP$, CH(08),
005930     AT (21,32),      RT
005950 "Press",
005970     AT (21,38), FAC(HEX(85)), ENTER$, CH(05),      RT
005990     AT (21,44),      RT
006010 "to print",
006030     AT (22,24),      RT
006050 "Press",
006070     AT (22,30), FAC(HEX(85)), PF14$, CH(04),      RT
006090     AT (22,35), "to select new attributes",      RT
006110     AT (23,19),      RT
006130 "Press",
006150     AT (23,25), FAC(HEX(84)), PF15$, CH(04),      RT
006170     AT (23,30),      RT
006190 "to select another Technical Word List",      RT
006210     AT (24,17),      RT
006230 "Press",
006250     AT (24,23), FAC(HEX(84)), PF16$, CH(04),      RT
006270     AT (24,28),      RT
006290 "to return to the Technical Word List Menu",      RT
006310 KEYS(BIN(0)&BIN(15)&BIN(16)&BIN(31)&BIN(32)&BIN(14)&BIN(30)),
006330 ON (BIN(15)&BIN(31)&BIN(16)&BIN(32)&BIN(14)&BIN(30)) GOTO NEW,NEW:
006350     ,EXIT,EXIT,YES,YES
006370     IF STR(TP$,1,1) = "P" THEN NEXT11
006390     IF STR(TP$,1,1) <> "P" THEN TERMPR
006410 NEXT11:     IF LPP% < 51 THEN NEXT13
006430     ER$="Lines per page' can be at most 50."
006450     GOTO PRINTSCR
006470
006490 IDERR1: PRINT

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 14:56 11/13/84 Page 7  
Input File is CRES@TPA in Library CRES@SRC on Volume CRES02

```

006510      PRINT BELL,HEX(94),ERR1$
006530      CLOSE #1
006550      FOR I=1 TO 200000 :NEXT I
006570      GOTO FILMNU
006590
006610  TERMPR:
006630  ER$="Print at ' has only legal values TERMINAL or PRINTER."
006650      GOTO PRINTSCR
006670
006690  REX: ER$="          'Starting word' comes after 'Ending word'."
006710      GOTO PRINTSCR
006750
006770  IOERR2:  PRINT
006790      PRINT HEX(94),"Error occurred on file "; FILE$
006810      PRINT HEX(94),"Error code = "; FS(#1)
006830      STOP "Touch ENTER to return to menu."
006850      GOTO CWMENU
006870
006890  NEXT13: IF STARTPH$="FIRST" OR ENDPH$="LAST" THEN FORD
006910      IF STARTPH$>ENDPH$ THEN REX
006930
006950  /* Now it's time to generate our print file name */
006970
006990  FORD:  DAT$= DATE
007010      D$= STR(DAT$,3,2)&"/"&STR(DAT$,5,2)&"/"&STR(DAT$,1,2)
007030      T$= STR(TIME,1,2)&":"&STR(TIME,3,2)
007050
007070      PFILE$="TAIRIB##"
007090      I#=0
007110  NEXT14: CONVERT I# TO STR(PFILE$,7,2),PIC(00)
007130      I#=I#+1
007150      CALL "LIMITS" (PFILE$,SLIB$,SVOL$,CODE$,MSG$)
007170      IF CODE$="Y" THEN NEXT14
007190  DISPLAY
007210      AT (01,03),
007230      "CRES: Print by Attribute Subsystem",
007250      AT (01,61),
007270      "(Generation Screen)",
007290      AT (02,03),
007310      "=====
007330      =====",
007350      AT (12,23),
007370      "Output file now being generated!"
007390      OPEN NOGETPARM #2,OUTPUT,FILE=PFILE$,LIBRARY=SLIB$,
007410      VOLU=SVOL$,SPACE=15000
007430      GOSUB TTL PAGE
007450      IF STARTPH$ = "FIRST" THEN STARTPH$ = ALL(HEX(00))
007470      IF ENDPH$ = "LAST" THEN ENDPH$ = ALL(HEX(FF))
007490      PAGE# = 1
007510      RECS# = 1
007530      READ #1, KEY=>STARTPH$,IOBUFF$
007550      IF STR(IOBUFF$,1,20)>ENDPH$ THEN HOME
007570      IF STR(IOBUFF$,1,16)=" TECHNICAL WORDS" THEN AGAN
007590  KNOT:  IF RTFL<1 THEN 7630
007610      IF STR(IOBUFF$,26,1)<>"*" THEN AGAN
007630      PSN= POS("JVCJNOLnApBa"= STR(IOBUFF$,21,1))
007650      IF T$(PSN)="X" THEN 7790

```

JM

JM

JM

RT

RT

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 14:56 11/13/84 Page 8  
Input File is CRES@TPA in Library CRES@SRC on Volume CRES02

```

007670      IF TS(14)=STR(IOBUFF$,22,1) THEN 7790      RT
007690      IF TS(15)=STR(IOBUFF$,24,1) THEN 7790      RT
007710      IF TS(16)=STR(IOBUFF$,23,1) THEN 7790      RT
007730      IF TS(17)=STR(IOBUFF$,27,1) THEN 7790      RT
007750      IF RTFL=2 AND STR(IOBUFF$,26,1)="*" THEN 7790      RT
007770      GOTO AGAN
007790      IF RECS% = 1 THEN GOSUB HEADER
007810      IF STR(TP$,1,1)="T" THEN BUG
007830      WRITE #2, USING IMG6,C$, " ",STR(IOBUFF$,1,20),
007850      AS(PSN),STR(IOBUFF$,22,1),STR(IOBUFF$,24,1),
007870      STR(IOBUFF$,23,1),STR(IOBUFF$,27,1)      RT
007890      IF(STR(IOBUFF$,26,1)<>"*")THEN BAG
007910      IF RTFL<>0 THEN BAG
007930      C$=HEX(0000)
007950      WRITE #2, USING IMG9, C$,STR(IOBUFF$,1,20)
007970      WRITE #2, USING IMG9, C$,STR(IOBUFF$,1,20)
007990      C$=HEX(0001)
008010      GOTO BAG
008030 BUG: IF STR(IOBUFF$,26,1)="*" THEN STR(IOBUFF$,26,1)="+"
008050      WRITE #2, USING IMG6,C$,STR(IOBUFF$,26,1),STR(IOBUFF$,1,20),
008070      AS(PSN),STR(IOBUFF$,22,1),STR(IOBUFF$,24,1),
008090      STR(IOBUFF$,23,1),STR(IOBUFF$,27,1)      RT
008110 BAG: RECS%=RECS%+1
008130      IF RECS%>LPP% THEN RECS%=1
008150      IF RECS%=1 THEN PAGE%=PAGE%+1
008170 AGAN: READ #1, IOBUFF$,EOD GOTO HOME
008190      IF STR(IOBUFF$,1,20)>ENDPH$ THEN HOME
008210      GOTO KNOT
008230 HOME: IF RECS%=1 THEN RECS%=51
008250      WRITE #2, USING FIT, C$
008270      RECS%=RECS%+2
008290      WRITE #2, USING DG, C$
008310      FOR I = RECS% TO LPP%+1
008330      WRITE #2, USING FIT, C$.
008350      NEXT I
008370      IF STR(TP$,1,1)="P" THEN HER1
008390      WRITE #2, USING IMG8, C$
008410      GOTO HER
008430 HER1: IF RTFL<>0 THEN HER: WRITE #2, USING RT, HEX(0002)
008450      C$=HEX(0000)
008470      FOR I=1 TO 2
008490      WRITE #2, USING RFI, C$
008510      NEXT I
008530      C$=HEX(0001)
008550 HER: CLOSE #2
008570      IF STR(TP$,1,1) = "P" THEN MESS
008590      CLOSE WS
008610      MSG2$ = "CANCEL Processing"
008630      MSK$=HEX(00)
008650      CMSG$ = " "
008670      CALL "LINK" ADDR("DISPFILE","P",RLIB$,RVOL$,3%,PFILE$,
008690      SLIB$,SVOL$, "C",MSG2$,27%, "H",MSK$,CMSG$,255%,C$,R%)      JM
008710      ER$ = " "      JM
008730      IF C%<>0 OR R%<>0
008750      THEN ER$ = STR(CMSG$,141)
008770      GOTO CMMENU
008790

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 14:56 11/13/84 Page 9  
Input File is CRES@TPA in Library CRES@SRC on Volume CRES02

```

008810 MESS: ER$="      File "&PPFILE$&" in Library "&SLIB$&" on "& : JM
008830 SVOL$ &" generated." JM
008850 GOTO BALONEY
008870
008890 NEW: PRINT PAGE, AT(12,28),HEX(84);"Closing last file"
008910 FOR I= 1 TO 50000 :NEXT I
008930 FILE$=" "
008950 CLOSE #1
008970 GOTO AGAIN
008990
009010 TFL_PGE:
009030 C$=HEX(8001)
009050 WRITE #2, USING TIF, C$
009070 C$ = HEX(0012)
009090 WRITE #2, USING TIF, C$
009110 C$ = HEX(0001)
009130 WRITE #2, USING TIF1, C$
009150 WRITE #2, USING TIF2, C$
009170 WRITE #2, USING TIF0, C$,D$,T$
009190 WRITE #2, USING TIF3, C$
009210 WRITE #2, USING TIF2, C$
009230 WRITE #2, USING TIF1, C$
009250 WRITE #2, USING TIF2, C$
009270 IF SBTITLE$ = " " THEN GERRY
009290 WRITE #2, USING TIF4, C$, SBTITLE$
009310 IF SBTITLE$<>" " THEN WRITE #2, USING :
009330 TIF10, C$, SBTITLE$
009350 WRITE #2, USING TIF2, C$
009370 GERRY: WRITE #2, USING TIF5, C$, FILE$
009390 WRITE #2, USING TIF2, C$
009410 WRITE #2, USING TIF6, C$, LIB$
009430 WRITE #2, USING TIF2, C$
009450 WRITE #2, USING TIF7, C$, IVOL$
009470 WRITE #2, USING TIF2, C$
009490 WRITE #2, USING TIF8, C$, STARTPH$
009510 WRITE #2, USING TIF2, C$
009530 WRITE #2, USING TIF9, C$, ENDPH$
009550 FOR I= 1 TO 17
009570 IF T$(I)="X" OR T$(I)="*" THEN FIRSTAT
009590 NEXT I
009610 FIRSTAT: WRITE #2,USING TIF2,C$: WRITE #2,USING TIF11,C$,A$(I)
009630 FOR K= I+1 TO 17
009650 IF T$(K)="X" OR T$(K)="*" THEN WRITE #2, USING TIF12, C$,A$(K)
009670 NEXT K
009690 WRITE #2, USING TIF2, C$
009710 WRITE #2, USING TIF1, C$
009730 RETURN
009750
009770 HEADER:
009790 IF PAGE#=1 THEN FIRST
009810 C$ = HEX(0001)
009830 IF STR(TP$,1,1)="P" THEN FRST1
009850 WRITE #2, USING TIF, C$
009870 WRITE #2, USING IMG8, C$
009890 GOTO FIRST
009910 FRST1: IF RTFL<>0 THEN FIRST
009930 WRITE #2, USING RT, HEX(0002)

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 14:56 11/13/84 Page 10  
Input File is CRES01PA in Library CRES0SRC on Volume CRES02

```

009950      C$=HEX(0000)
009970      FOR I=1 TO 2
009990      WRITE #2, USING RT1, C$
010010      NEXT I
010030 FIRST: C$ = HEX(8001)
010050      WRITE #2, USING FIT, C$
010070      C$ = HEX(0001)
010090      IF SBTITLE$ = " "
010110      THEN WRITE #2, USING IMG,C$,D$,T$,PAGE%
010130      ELSE GOTO FOUT
010150      GOTO ALRITE
010170 FOUT:  WRITE #2, USING IMG1,C$,SBTITLE$,D$,T$,PAGE%
010190      IF SBTITL1$<>"
010210      THEN WRITE #2, USING IMG7,C$,SBTITL1$
010230 ALRITE: WRITE #2, USING IMG2, C$, FILE$, LIB$, IVOL$
010250      WRITE #2, USING IMG3, C$
010270      WRITE #2, USING IMG4, C$
010290      WRITE #2, USING IMG5, C$
010310      WRITE #2, USING IMG3, C$
010330      RETURN
010350
010370 TIT0:###      : #####
010390      #####      :
010410 TIT: ###
010430 DG:###      !!! END OF LISTING !!!
010450 TIT1:###      :-----
010470 -----
010490 TIT2:###      :
010510      :
010530 TIT3:###      : ***** TECHNICAL WORD LIST ***** RT
010550 *      :
010570 TIT4:###      : TITLE OF LIST: #####
010590 #####      :
010610 TIT5:###      : , INPUT FILE: #####
010630      :
010650 TIT6:###      : INPUT LIBRARY: #####
010670      :
010690 TIT7:###      : INPUT VOLUME: #####
010710      :
010730 TIT8:###      : STARTING WORD: #####
010750 #####      :
010770 TIT9:###      : ENDING WORD: #####
010790 #####      :
010810 TIT10:###      : #####
010830 #####      :
010850 TIT11:###      : ATTRIBUTES SPECIFIED: #####
010870 #####      :
010890 TIT12:###      : #####
010910 #####      :
010930 IMG:### #####      :
010950      PAGE #####
010970 IMG1:### TITLE OF LIST : #####
010990 #####      PAGE #####
011010 IMG7:### #####
011030 IMG2:### Input File is ##### in Library ##### on Volume ###
011050 ###
011070 IMG3:### -----

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 14:56 11/13/84 Page 11  
Input File is CRES@PPA in Library CRES@SRC on Volume CRES02

```

011090 =====
011095 IMG4:### WORD Part of Past To Be N! RT
011100 egative Mand. Tech. RT
011105 IMG5:### Speech Participle ! RT
011110 Word RT
011115 IMG6:### # ##### # # ! RT
011120 # # RT
011210 IMG9:### #####
011230 IMG8:### + = ROOT WORD
011250 RT:### BOLDFACE = Root word
011270 RT1:### BOLDFACE
011290
011310 /* THIS FUNCTION PRINTS OUT ERROR MESSAGES AND */
011330 /* SETS UP FLASHING FAC'S IN APPROPRIATE PLACES */
011350 DEF FN'7(MSG$,B$,C$,D$)
011370 PRINT
011390 M$=HEX(94)
011410 G$=B$
011430 L$=C$
011450 V$=D$
011470 RETURN
011490
011510 EXIT: CLOSE #1
011530 OUT: /* RETURNING TO TECHNICAL WORD LIST MENU */

```

JM

Technical Report 86-001

THIS PAGE INTENTIONALLY LEFT BLANK.

## WORD AND PHRASE SUBSTITUTE LIST SUBSYSTEM

### Create/Maintain Word and Phrase Substitute List Menu (CRES@WSL)

CRES@WSL in library CRES@PRC is the Procedure Language menu routine that serves as the driver for the Create/Maintain Word and Phrase Substitute Lists Subsystem. It enables the user to select by PF key among options to create/maintain the Word Substitute List or the Phrase Substitute List, to print the Word Substitute List, the Phrase Substitute List, or a combined Word and Phrase Substitute List, or to return to the CRES Master Menu.

A complete listing follows.

Technical Report 86-001

THIS PAGE INTENTIONALLY LEFT BLANK.

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 16:12 11/13/84 Page 1  
Input File is CRES@WSL in Library CRES@PRC on Volume CRES01

```

*
*      PROCEDURE      CRES SUBSTITUTION LIST MENU
*
*      AUTHOR      ROSE TANK
*      DATE      10/06/83
*
*      Abstract: This procedure is a subsystem of CRES Master Menu
*                (option 3 - Create/Maintain Substitution Lists).
*
*      It allows the user to select and run any
*      substitution list file option.
*
*      DECLARE      &USERINPT AS INTEGER      INITIAL 0
*      DECLARE      &OPTIONERR AS STRING (79) INITIAL " "
*      DECLARE      &OPTIONERR2 AS STRING (79) INITIAL " "
*      DECLARE      &USERNAME AS STRING (24) INITIAL " "
*      DECLARE      &ERRMSG AS STRING (34) INITIAL " "
*      DECLARE      &RUNLIB AS STRING (8) INITIAL " "
*      DECLARE      &RUNVOL AS STRING (6) INITIAL " "
*
*      EXTRACT &USERNAME=USERNAME, &RUNLIB=OUTLIB, &RUNVOL=RUNVOL
*
*      GETUSERINPT:  PROMPT PFKEY = &USERINPT      CENTER
*      DIM "CRES: Substitution Subsystem      (Substi RT
*      tution Menu Screen)";
*      DIM "=====
*      =====";
*      DIM " ";
*      DIM " Use the appropriate",BRIGHT " PF key", " to select the desi
*      red operation.";
*      DIM " ";
*      DIM "      ",LINE "PF",DIM " ";
*      LINE "      Operation      ";
*      DIM " ";
*      DIM "      ",BRIGHT "1", " - Create/Maintain Word Substi
*      tution List";
*      DIM "      ",BRIGHT "2", " - Create/Maintain Phrase Suos
*      titution List";
*      DIM "      ",Bright "3", " - Print Word Substitution Lis
*      t";
*      DIM "      ",Bright "4", " - Print Phrase Substitution L
*      ist";
*      DIM "      ",Bright "5", " - Print Combination of word a
*      nd Phrase Substitutes";
*      DIM " ";
*      DIM "      ",Bright "6", " - Return to CRES Master Menu"
*      ;
*      DIM " ";
*      BRIGHT &OPTIONERR;
*      BLINK &OPTIONERR2;
*
*      ASSIGN      &ERRMSG = " "
*      ASSIGN      &OPTIONERR = " "
*      ASSIGN      &OPTIONERR2 = " "
*
*      IF      &USERINPT EQ 16      GOTO USEREXIT
*      IF      &USERINPT EQ 32      GOTO USEREXIT
*      IF      &USERINPT EQ 1      GOTO RUNWSL
*      IF      &USERINPT EQ 17      GOTO RUNWSL

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 16:12 11/13/84 Page 2  
Input File is CRES@WSL in Library CRES@PRC on Volume CRES01

```

IF &USERINPT EQ 2          GOTO RUNPSL          005110
IF &USERINPT EQ 18         GOTO RUNPSL          005120
IF &USERINPT EQ 3          GOTO RUNPWS          RT 005200
IF &USERINPT EQ 19         GOTO RUNPWS          RT 005300
IF &USERINPT EQ 4          GOTO RUNPPS          005330
IF &USERINPT EQ 20         GOTO RUNPPS          005340
IF &USERINPT EQ 5          GOTO RUNCWP          005350
IF &USERINPT EQ 21         GOTO RUNCWP          005360
                                005400
                                ASSIGN &OPTIONERR = "I'm sorry, but your selected option
                                is not currently available." 005500
                                005600
                                ASSIGN &OPTIONERR2 = "Please select another topic." 005700
                                GOTO GETUSERINPT 005800
*                                005900
RUNWSL:  RUN CRES@WSL IN &RUNLIB ON &RUNVOL      006000
          IF RUNWSL EQ 0      GOTO GETUSERINPT 006100
          GOTO USEREXIT      006200
*                                006300
RUNPWS:  RUN CRES@WSP IN &RUNLIB ON &RUNVOL      RT 006400
          IF RUNPWS EQ 0      GOTO GETUSERINPT RT 006500
          GOTO USEREXIT      006600
*                                006710
RUNPSL:  RUN CRES@PSL IN &RUNLIB ON &RUNVOL      006720
          IF RUNPSL EQ 0      GOTO GETUSERINPT 006730
          GOTO USEREXIT      006740
*                                006750
RUNPPS:  RUN CRES@PSP IN &RUNLIB ON &RUNVOL      006760
          IF RUNPPS EQ 0      GOTO GETUSERINPT 006770
          GOTO USEREXIT      006780
*                                006782
RUNCWP:  RUN CRES@SBP IN &RUNLIB ON &RUNVOL      006784
          IF RUNCWP EQ 0      GOTO GETUSERINPT 006786
          GOTO USEREXIT      006788
*                                006790
USEREXIT: RETURN                                006800

```

Create/Maintain Word Substitute List (CRES@WSL)

CRES@WSL in library CRES@SRC is the BASIC language program to create/maintain the Word Substitute List. The object language version is CRES@WSL in library CRES@SAV.

This program first allows the user to specify the Word Substitute List to be created or maintained. Then, the program enables the user to add, delete, or modify any given word and its substitute(s) and attributes. Up to two substitutes can be specified. The first substitute can contain up to 20 characters; the second substitute can contain up to 15 characters.

The only characters accepted in the input word and its substitute(s) are the letters A through Z, ".", "-", and "'". There can be no embedded blanks. The data record for each word also contains information about the word's attributes: the part of speech (see table 1), the past participle flag, and the negative flag. A complete data record description is provided in appendix A.

There is a protection system implemented for this list, and the program makes provision for clearing the protect bytes, if for some reason the bytes have been left set incorrectly.

A complete program listing follows. Appendix C contains a sample flowchart.

Technical Report 86-001

THIS PAGE INTENTIONALLY LEFT BLANK.

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 15:05 11/13/84 Page 1  
Input File is CRES0WSL in Library CRES0SRC on Volume CRES02

```

000010 /* This is the CRES WORD SUBSTITUTION MAINTENANCE SUBSYSTEM */ RT
000110 /* */ RT
000210 /* AUTHOR: Rose Tank */ JM
000310 /* Date: October 12, 1983 */ RT
000410 /* */ RT
000510 RT
000610 /* The following code sets up a screen, which asks for the */ RT
000710 /* name of the file containing the word sub. list that is */ RT
000810 /* to be modified. It also provides the opportunity to change */ RT
000910 /* the library and volume names which are set to the appro- */ RT
001010 /* priate defaults through the "Extract" system routine. */ RT
001110 RT
001210 RT
001310 DIM WORD$20,ERR1$50,CODE$1,MSG$50,P$40,G$1,PROTECT$3 RT
001410 DIM F1$20,IOBUFF$61,ER1$50,XYZ$1,MSG$21,P$(13)3,FIL$22 RT
001510 DIM PFK$4,TAB$3,BTAB$7,FILE$8,USER$3,PF1$3,USERNAME$24,WRKST$3 RT
001610 DIM LIB$ 08,PASSWRD$13,X$1 JM
001710 DIM IVOL$ 06,HEADER$50,L$1,V$1,B$1,C$1,D$1,M$1 RT
001810 DIM MOD$ 15,SUBONE$20,SUBTWO$15,PPFL$1,PRTO$PC$1 RT
001910 DIM DEF$ ' 07,NEGFL$1 RT
002010 DIM LEG$ 05,A$10,A7$7 RT
002110 DIM NOEXIST$68, E$1 RT
002210 DIM ER$ 68 RT
002310 DIM CAP$25 RT
002410 HEADPR$ = "CRES WORD SUBSTITUTION MAINTENANCE SUBSYSTEM" RT
002510 FIL$ = "Word Substitution List" RT
002610 MSG$ = "Parts of speech codes" RT
002710 TAB$ = "TAB": BTAB$ = "BACKTAB" RT
002810 P$(1) = "(J)": P$(2) = "(j)": P$(3) = "(A)": P$(4) = "(a)": P$(5) = "(B)" RT
002910 P$(6) = "(C)": P$(7) = "(c)": P$(8) = "(N)": P$(9) = "(n)": P$(10) = "(P)" RT
003010 P$(11) = "(O)": P$(12) = "(V)": P$(13) = "(d)" RT
003110 FILE$ = "PART OF SPEECH" JM
003210 FILE6$ = "DEFINITION" JM
003310 ENTER$ = "ENTER" RT
003410 PF1$ = "PF1": PF16$ = "PF16" RT
003510 RT
003610 F1$ = HEX(A014A014A00FA001A001A001A0030013) RT
003710 F$ = "Filename not found on (library/volume)." RT
003810 ERR1$ = "I/O ERROR - Please check filename and respecify." RT
003910 RT
004010 CALL "EXTRACT" ADDR("ID",USER$,"IL",LIB$,"IV",IVOL$) JM
004110 DEF=0 RT
004210 FILE$ = "WRDSBLIST" JM
004310 AGAIN: G$ = HEX(81) RT
004410 MSG$ = " ": L$ = HEX(81) RT
004510 F$ = HEX(81): M$ = HEX(80) RT
004610 RT
004710 FILE2$: RT
004810 ACCEPT : RT
004910 AT (01,01), : RT
005010 "MSG: Word Substitution List Maintenance Subsystem", : RT
005110 AT (01,57), : RT
005210 "(File Selection Screen)", : RT
005310 AT (02,01), : RT
005410 " : RT
005510 ===== : RT
005610 AT (04,12), : RT

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 15:05 11/13/84 Page 2  
Input File is CRES@WSL in Library CRES@SRC on Volume CRES02

```

005710 "Welcome to the",           : RT
005810         AT (04,27), FAC(HEX(85)), : RT
005910 HEADER$,                       : RT
006010         AT (07,06),               : RT
006110 "Please specify the",           : RT
006210         AT (07,25), FAC(HEX(84)), : RT
006310 FIL$,                             : RT
006410         AT (07,48),               : RT
006510 "you want to modify or create.", : RT
006610         AT (13,30),               : RT
006710 "File      =",                  : RT
006810         AT (13,43), FAC(G$), FILE$ , CH(08), : RT
006910         RANGE ( "!" " " , "XXXXXXXXXX"), : RT
007010         AT (14,30),               : RT
007110 "Library  =",                  : RT
007210         AT (14,43), FAC(L$), LIB$ , CH(08), : RT
007310         RANGE ( "!" " " , "XXXXXXXXXX"), : RT
007410         AT (15,30),               : RT
007510 "Volume   =",                  : RT
007610         AT (15,43), FAC(V$), IVOL$ , CH(06), : RT
007710         RANGE ( "!" " " , "XXXXXXXXXX"), : RT
007810         AT (18,24), FAC(M$), MSG$, : RT
007910         AT (20,30), "Press",      : RT
008010         AT (20,36), FAC(HEX(84)), : RT
008110 ENTER$,                           : RT
008210         AT(20,42),                : RT
008310 "to continue",                   : RT
008410         AT (21,17), "Press",      : RT
008510         AT (21,23), FAC(HEX(84)), : RT
008610 PF16$,                             : RT
008710         AT (21,28),               : RT
008810 "to return to the Substitution List Menu", : RT
008910 ON (BIN(16)&BIN(32)) GOTO OUT,OUT : RT
009110                                     : RT
009210 /* SET UP THE USER FILE BLOCK */ : RT
009310                                     : RT
009410 SELECT #1, "WROSUB", VARC, INDEXED, RECSIZE=80, KEYPOS=1, KEYLEN=20, : JM
009510         IOERR GOTO IOERR1         : RT
009610                                     : RT
009710 /* SEE IF THE SPECIFIED FILE EXISTS */ : RT
009810                                     : RT
009910 CALL "LIMITS"(FILE$, LIB$, IVOL$, CODE$, MSG$) : RT
010010 IF CODE$="Y" THEN GOTO ZZZ         : RT
010110 IF CODE$="N" THEN CREATE           : RT
010210 IF CODE$="V" THEN GOSUB 7(STR(MSG$,3), HEX(81), HEX(81), HEX(91)) : RT
010310 IF CODE$="L" THEN GOSUB 7(STR(MSG$,3), HEX(81), HEX(91), HEX(81)) : RT
010410 IF CODE$="E" THEN GOSUB 7(STR(MSG$,3), HEX(81), HEX(81), HEX(81)) : RT
010510 GOTO FILMNU                         : RT
010610                                     : RT
010710 /* IF WE FALL HERE, IT MUST BE TIME TO OPEN FILE */ : RT
010810 CREATE:                             : RT
010910 ACCEPT                               : RT
011010 "CRES: Word Substitution List Maintenance Subsystem", : JM
011110         AT (01,60),               : JM
011210 "(Create File Screen)",           : JM
011310         AT (02,01),               : RT
011410 "=====":                          : RT

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 15:05 11/13/84 Page 3  
Input File is CRES@WSL in Library CRES@SRC on Volume CRES02

```

011510 =====, : RT
011610 AT (06,08), : RT
011710 "The file you have selected does not currently exist in the liora: RT
011810 ry", : RT
011910 AT (07,08), : RT
012010 "and volume specified. For your convenience, the following are ", : JM
012110 AT (08,08), : RT
012210 "the parameters you submitted.", : RT
012310 AT (11,30), : RT
012410 "File =", : RT
012510 AT (11,40),FAC(HEX(85)),FILE$ , CH(08), : RT
012610 AT (12,30), : RT
012710 "Library=", : RT
012810 AT (12,40), FAC(HEX(85)), LIB$ , CH(08), : RT
012910 AT (13,30), : RT
013010 "Volume =", : RT
013110 AT (13,40), FAC(HEX(85)), IVOL$ , CH(06), : RT
013210 AT (17,15), : RT
013310 "To create a new file from these parameters, press", : JM
013410 AT (17,65), FAC(HEX(85)), ENTER$ , CH(05), : RT
013510 AT (19,26), : RT
013610 "To select another file, press", : JM
013710 AT (19,56), FAC(HEX(85)), PFI$ , CH(03), : RT
013810 KEYS(BIN(0)&BIN(1)&BIN(17)),KEY(OPTION), : RT
013910 ON (BIN(1)&BIN(17)) GOTO FILMNU,FILMNU : RT
014010 : RT
014110 OPEN NOGETPARM #1, OUTPUT, FILE= FILE$, LIBRARY=LIB$, : RT
014210 VOLUME= IVOL$, SPACE=1000 : RT
014310 IOBUFF$= " WORD SUBSTITUTION" : RT
014410 WRITE #1, IOBUFF$ : RT
014510 CLOSE #1 : RT
014610 ZZZ: CALL "READFOR" ADDR(FILE$,LIB$,IVOL$,0%,"RS",RSIZ%,RET%) : JM
014710 IF RSIZ%<>80 THEN ZZ1 : JM
014810 OPEN NOGETPARM #1,SHARED, FILE= FILE$, LIBRARY = LIB$, : JM
014910 VOLUME = IVOL$ : RT
015210 READ #1, IOBUFF$ : RT
015310 IF IOBUFF$=" WORD SUBSTITUTION" THEN YES : RT
015410 CLOSE #1 : JM
015510 ZZ1: M$=HEX(94) : JM
015610 MSG$= ERR1$ : RT
015710 GOTO FILMNU : RT
015810 YES: : RT
015910 /* This section of code generates another screen, which asks */ JM
016010 /* for the particular word and its corresponding operation - */ JM
016110 /* either insert word, delete word, or modify word. The user is */ RT
016210 /* also at this point given the option of returning to the */ JM
016310 /* Substitution List Menu, if desired. */ JM
016410 : RT
016510 ZAP$="Please select the word:" : RT
016610 WORD$=" " : RT
016710 WORDEQ$="word = " : RT
016810 PFK$=" PF " : RT
016910 OPER$=" Operation" : RT
017010 O$="1" : RT
017110 T$="2":TH$="3":Q$="14":R$="15":S$="16" : RT
017210 ER$ = " " : RT
017310 ER1$=" " : RT

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 15:05 11/13/84 Page 4  
Input File is CRES@WSL in Library CRES@SRC on Volume CRES02

```

017320 PASSWRD$=HEX(000102030E0F101112131E1F20) JM
017410 NOEXIST$="The word you are attempting to modify does not exist in: RT
017510 this list." RT
017520 X$=HEX(81) JM
017610 E$=HEX(84) RT
017710 RT
017810 CWMENU: RT
017910 ACCEPT : RT
018010 AT (01,01), JM
018020 "CRES: Word Substitution List Maintenance Subsystem", : JM
018110 AT (01,59), "(Word & Option Screen)", : JM
018210 AT (02,01), "===== RT
018310 =====", : RT
018410 AT (04,08), FAC(HEX(8C)), : RT
018510 ZAP$, : RT
018610 AT (06,28), FAC(HEX (8C)), : RT
018710 WORDPQ$, : RT
018810 AT (06,36), FAC(X$), WORD$ , CH(20), JM
018910 AT (08,13), FAC( E$), ER$, : RT
019010 AT (09,16), FAC( E$), ER$, : RT
019110 AT (11,08), "To cont: RT
019210 inue, select the PF key corresponding to the desired operation.", RT
019310 AT (13,30), FAC(HEX(A5)), : RT
019410 PFK$, : RT
019510 AT (13,38), FAC(HEX(AC)), : RT
019610 OPER$, : RT
019710 AT (15,31), FAC(HEX(84)), : RT
019810 Q$, : RT
019910 AT (15,39), : RT
020010 "INSERT word", : RT
020110 AT (16,31), FAC(HEX(84)), : RT
020210 T$, : RT
020310 AT (16,39), : RT
020410 "DELETE word", : RT
020510 AT (17,31), FAC(HEX(84)), : RT
020610 TH$, : RT
020710 AT (17,39), : RT
020810 "MODIFY word", : RT
020910 AT (19,31), FAC(HEX(84)), : RT
021010 Q$, : RT
021110 AT (19,39), : RT
021210 "Protection Override", : RT
021310 AT (20,31), FAC(HEX(84)), : RT
021410 R$, : RT
021510 AT (20,39), : RT
021610 "Return to Word Substitution List Selection", : RT
021710 AT (21,31), FAC(HEX(84)), : RT
021810 S$, : RT
021910 AT (21,39), : RT
022010 "Return to Substitution List Menu", : JM
022110 KEYS(PASSWRD$), : JM
022310 KEY(OPTION) JM
022420 IF OPTION=15 OR OPTION=31 THEN NEW JM
022430 IF OPTION=16 OR OPTION=31 THEN EXIT JM
022510 RT
022610 FOR I=1 TO LEN(WORD$) JM
022710 IF POS("A B C D E F G H I J K L M N O P Q R S T U V W X Y Z -"=STR(WORD$,I,1))=0 : JM

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 15:05 11/13/84 Page 5  
Input File is CRES@WSL in Library CRES@SRC on Volume CRES02

```

022810                                THEN ILLERR                                JM
022910 NEXT I                                                                JM
023010
023110 E$=HEX(8D)                                                            RT
023210 ER$=" "                                                                RT
023310 ER1$=" "                                                                RT
023410 IF (OPTION=0)AND(DP=1) THEN DEL                                        RT
023510 IF DP=1 THEN UPDATE                                                    RT
023610 IF (OPTION=14) OR (OPTION=30) THEN OVERRIDE                            RT
023710 IF STR(WORD$,1,1)<>" " THEN GOTO REED                                RT
023810                                                                        RT
023910 PRINT BELL                                                            RT
024010 ER$=" The first character cannot be blank."                            RT
024110 ER1$=" Please reenter the appropriate word and PF key."                RT
024210 E$= HEX(94)                                                            RT
024310 GOTO CWMENU                                                            RT
024410                                                                        RT
024510 ILLERR:                                                                JM
024610 ER$="Illegal character found in word - please reenter."                JM
024710 E$=HEX(94): GOTO CWMENU                                                JM
024810
024910 OVERRIDE:                                                            RT
025010 IF NF=1 AND P%1 THEN CHK                                              RT
025110 ER$="Please check if word exists first by selecting an option."        RT
025210 E$=HEX(84)                                                            RT
025310 GOTO CWMENU                                                            RT
025410                                                                        RT
025510 CHK:                                                                RT
025610 IF USER$=PROTECT$ THEN REMOVE                                        RT
025710 ER$="You don't have the clearance to override the protect."            RT
025810 ER1$=" "                                                                RT
025910 E$= HEX(84)                                                            RT
026010 GOTO CWMENU                                                            RT
026110                                                                        RT
026210 REMOVE: PROTECT$=" " :P%0                                            RT
026310 READ #1, HOLD, KEY=WORD$,IOBUFF$                                       RT
026410 $PACK (F=FI$) IOBUFF$ FROM WORD$, SUBONE$,SUBTWO$, : RT
026510 PR10SPC$,PPFL$,NEGFL$,PROTECT$                                         RT
026610 REWRITE #1, IOBUFF$, DATA GOTO IOERR2                                RT
026710 ER$=" You may now select your option."                                RT
026810 ER1$=" "                                                                RT
026910 E$= HEX(84)                                                            RT
027010 GOTO CWMENU                                                            RT
027110                                                                        RT
027210 IOERR1: PRINT                                                         RT
027310 M$=HEX(94)                                                            RT
027410 MSG$= ERR1$                                                            RT
027510 CLOSE #1                                                              RT
027610 GOTO FILMNU                                                            RT
027710                                                                        RT
027810 IOERR2: PRINT                                                         RT
027910 PRINT HEX(94),"Error occurred on file "; FILE$                        RT
028010 PRINT HEX(94),"Error code = "; FS(1)                                  RT
028110 STOP "Touch ENTER to return to menu."                                RT
028210 GOTO CWMENU                                                            RT
028310                                                                        RT
028410 REED: IF OPTION<>0 THEN SRCH                                          RT

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 15:05 11/13/84 Page 6  
Input File is CRES@WSL in Library CRES@SRC on Volume CRES02

```

028510 ER$= " Command missing." RT
028610 ES= HEX(94) RT
028710 GOTO CWMENU RT
028715 RT
028720 SRCH: P%=1 RT
028810 READ #1,HOLD,KEY=WORD$,IOBUFF$,EOD GOTO NOTFOUND, : RT
028910 DATA GOTO IOERR2 RT
029010 RT
029110 NF=1 RT
029210 RT
029310 /* Unpack and check if currently being modified by another */ RT
029410 RT
029510 $UNPACK (F=F1$) IOBUFF$ TO WORD$,SUBONE$,SUBTWO$, : RT
029610 PRIO$PC$,PPFL$,NEGFL$,PROTECT$ RT
029710 IF PROTECT$= " " THEN NOW RT
029810 CALL "FINDUSER" ADDR(PROTECT$,USERNAMES$,WRKST$) RT
029910 CONVERT WRKST$ TO WRKST$, PIC(###) RT
030010 ES= HEX(94) RT
030110 ER$= " Sorry, "& USERNAMES$ & " is currently editing " RT
030210 'ER1$= " "& WORD$ & "'& " at workstation "& WRKST$ RT
030310 REWRITE #1, IOBUFF$ RT
030410 GOTO CWMENU RT
030510 RT
030610 NOW: PROTECT$= USER$ RT
030710 $PACK (F=F1$) IOBUFF$ FROM WORD$,SUBONE$,SUBTWO$, : RT
030810 PRIO$PC$,PPFL$,NEGFL$,PROTECT$ RT
030910 REWRITE #1, IOBUFF$, DATA GOTO IOERR2 RT
031010 RT
031110 /* NOW BRANCH TO THE OPTION SELECTED */ RT
031210 OPT: ON OPTION GOTO INSERT,DELET,MODIFY RT
031310 ON OPTION-16 GOTO INSERT,DELET,MODIFY RT
031410 RT
031510 NOTFOUND: NF=0 RT
031610 GOTO OPT RT
031710 RT
031810 INSERT: RT
031910 /* NF=0 indicates the word doesn't yet exist so go ahead */ RT
032010 /* and insert. NF=1 indicates the word exists. */ RT
032110 RT
032210 IF NF=1 THEN EXIST RT
032310 MOD$= "(Insert Screen)" RT
032320 MESING$="inserting word" JM
032410 SUBONE$,SUBTWO$,PRIO$PC$,PPFL$,NEGFL$=" " RT
032510 PROTECT$= USER$ RT
032610 $PACK (F=F1$) IOBUFF$ FROM WORD$, SUBONE$,SUBTWO$, : RT
032710 PRIO$PC$,PPFL$,NEGFL$,PROTECT$ RT
032810 WRITE #1, IOBUFF$, DATA GOTO IOERR2 RT
032910 GOTO INS RT
033010 RT
033110 /* Coming from insert with the word currently existing */ JM
033310 RT
033410 EXIST: RT
033510 ER$= " The word "& WORD$ & " already exists in this list." RT
033610 ES= HEX(94) RT
033710 READ #1, HOLD, KEY= WORD$,IOBUFF$ RT
033810 PROTECT$= " " :P%=0 RT
033910 $PACK (F=F1$) IOBUFF$ FROM WORD$, SUBONE$,SUBTWO$, : RT

```

# Technical Report 86-001

wang VS Integrated Editor - Version 6.10.00 15:05 11/13/84 Page 7  
Input File is CRES@WSL in Library CRES@SRC on Volume CRES02

```

034010          PRTOSPC$,PPFL$,NEGFL$,PROTECT$      RT
034110          REWRITE #1, IOBUFF$, DATA GOTO IDERR2 RT
034210          GOTO CWMENU                          RT
034310                                              RT
034410 DELET: IF NF=0 THEN NODEL                    RT
034510          DF=1                                  RT
034610          ER$= "          Press ENTER to confirm deletion of "& WORD$ RT
034710          ER1$="          or press PF14 to cancel the deletion." RT
034720          X$=HEX(84)                            JM
034730          PASSWRD$=HEX(000E1E)                  JM
034810          E$= HEX(94)                          RT
034910          GOTO CWMENU                          PT
035010                                              RT
035110 DEL:   READ #1, HOLD,KEY=WORD$,IOBUFF$      RT
035210          DELETE #1                            RT
035310          DF=0                                  RT
035410          ER$= "          "&WORD$&" deleted - Please select your next word." RT
035510          ZAP$=" "                                RT
035520          X$=HEX(81)                            JM
035530          PASSWRD$=HEX(000102030E0F101112131E1F20) JM
035610          E$= HEX(8C)                          RT
035710          WORD$=" "                            RT
035720          NF=0                                  JM
035810          GOTO CWMENU                          RT
035910                                              RT
036010 NODEL: ER$="          The word "&WORD$&" does not exist in this list." RT
036110          E$= HEX(94)                          RT
036210          GOTO CWMENU                          RT
036310                                              RT
036410 MODIFY: IF NF=1 THEN GETIT                   RT
036510          ER$ = NOEXISTS$ /* if word not in list, */ RT
036610          E$=HEX(94) /* set up message and fac and */ RT
036710          GOTO CWMENU /* go back up to previous screen */ RT
036810                                              RT
036910 GETIT:                                         JM
037210                                              RT
037310 MOD$      ="(Modify Screen)"                  RT
037320 MESING$="modifying word"                     JM
037410 INS:                                           RT
037510 DEF$      ="CURRENT"                          RT
037610 A5$      ="ATTRIBUTES"                        RT
037710 LEG$     ="LEGAL"                            RT
037810 A7$      ="ENTRIES"                          RT
037910 ER1$     =" "                                RT
038010 XYZ$     ="HEX(8C)"                          RT
038110 ERONINS:                                       RT
038210 ACCEPT                                         ! RT
038310          AT (01,01),                            ! RT
038410 "CRES: Word Substitution List Maintenance Subsystem", ! JM
038510          AT (01,66), FAC(HEX(8C)),MOD$,          ! RT
038610          AT (02,01), "=====RT
038710          "=====RT
038810          AT (03,07),                            ! RT
038910 "For an explanation of legal entries, press",    ! JM
039010          AT (03,50),FAC(HEX(84)), PF1$,          ! RT
039110          AT (03,54),                            ! RT
039210 "; otherwise, enter",                          ! RT

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 15:05 11/13/84 Page 8  
Input File is CRES@WSL in Library CRES@SRC on Volume CRES02

```

039310      AT (04,07),                                : RT
039410 "the appropriate attribute(s).",                : RT
039510      AT (07,31),FAC(HEX(AD)), DEF$,              : RT
039610      AT (07,39),FAC(HEX(AD)), A5$,              : RT
039710      AT (07,58),FAC(HEX(AD)), LEX$,             : RT
039810      AT (07,64),FAC(HEX(AD)), A7$,              : RT
039910      AT (09,07),                                : RT
040010 "WORD:",                                       : RT
040110      AT (09,15),FAC(HEX(84)), WORD$,              : RT
040210      RANGE ( ":", " ", "#####"),              : RT
040310      AT (11,07),                                : RT
040410 "FIRST SUBSTITUTE:",                          : RT
040510      AT (11,30), SUBONE$,CH(20),                  : RT
040610      AT (12,07),                                : RT
040710 "SECOND SUBSTITUTE:",                        : RT
040810      AT (12,30), SUBTWO$,CH(20),                  : RT
040910      AT (14,07),                                : RT
041010 "PART OF SPEECH:",                          : RT
041110      AT (14,41),FAC(HEX(80)),PRTO$PC$, CH(01),    : RT
041210      AT (14,57),                                : RT
041310 "(ABCDEFGHIJOPVadjn)",                          : RT
041410      AT (15,07),                                : RT
041510 "PAST PARTICIPLE:",                          : RT
041610      AT (15,41), PPFI$, CH(01),                  : RT
041710      AT (15,57),                                : RT
041810 "(* or NO ENTRY)",                              : RT
041910      AT (16,07),                                : RT
042010 "NEGATIVE:",                                : RT
042110      AT (16,41), NEGFI$, CH(01),                  : RT
042210      AT (16,57),                                : RT
042310 "(* or NO ENTRY)",                              : RT
042410      AT (18,18), FAC(XYZ$), ER$,                  : RT
042510      AT (21,15),"Use",FAC(HEX(85)),PA3$,"or",FAC(HEX(86)), : RT
042610      BTAB$,"to move between modifiable fields.", : RT
042710      AT (23,30),                                : RT
042810 "Press",                                       : RT
042910      AT (23,36),FAC(HEX(84)),ENTER$,              : RT
043010      AT (23,42),                                : RT
043110 "to continue",                                : RT
043210      AT (24,21),                                : RT
043310 "Press",AT(24,27),FAC(HEX(84)),PP16$,AT(24,32), : RT
043410 "to exit without ",                            : RT
043420      AT (24,48), FAC(HEX(8C)), MESING$,          : RT
043510 KEYS(AIN(0)&BIN(16)&BIN(32)&BIN(1)&BIN(17)),KEY(OPTION), : RT
043610 ON (BIN(16)&BIN(32)) GOTO EXMOD,EXMOD           : RT
043710                                             : RT
043810 IF (OPTION=1) OR (OPTION=17) THEN DEFIN         : RT
043910                                             : RT
044010 IF SUBONE$<>" " THEN CHKREST                   : RT
044110 ER$="          The First Substitute can not be blank." : RT
044210 XYZ$= HEX(94)                                : RT
044310 GOTO ERONINS                                   : RT
044410                                             : RT
044510 /* Check for validity of entries */             : RT
044610                                             : RT
044710 CHKREST: IF POS("ABCDEFGHIJOPVadjn")=PRTO$PC$>0 THEN NEXT1 : RT
044810 ER$="          Part of Speech value is illegal."   : RT

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 15:05 11/13/84 Page 9  
Input File is CRES@WSL in Library CRES@SRC on Volume CRES02

```

044910      XYZ$ = HEX(94)                      RT
045010      GOTO ERONINS                        RT
045110      RT                                  RT
045210 NEXT1: IF POS(" **=PPFL$)>0 THEN NEXT2    RT
045310      ER$ = "      Past Participle value is illegal." RT
045410      XYZ$ = HEX(94)                      RT
045510      GOTO ERONINS                        RT
045610 NEXT2: IF POS(" **=NEGFL$)>0 THEN UPDATE    RT
045710      ER$ = "      Negative value is illegal." RT
045810      XYZ$ = HEX(94)                      RT
045910      GOTO ERONINS                        RT
046010      RT                                  RT
046110 EXMOD: READ #1, HOLD, KEY=WORD$, IOBUFF$   RT
046210      IF NF=1 THEN NOCHAN                  RT
046310      DELETE #1                            RT
046410      ER$= "      The word "&WORD$&" was not inserted." RT
046510      GOTO CWMENU                          RT
046610      RT                                  RT
046710 NOCHAN: PROTECT$ = " " :P$=0              RT
046810      $PACK (P=F1$) IOBUFF$ FROM WORD$,SUBONE$,SUBTWO$, : RT
046910      PROTOSPC$,PPFL$,NEGFL$,PROTECT$      RT
047010      REWRITE #1,IOBUFF$,DATA GOTO IOERR2  RT
047110      ER$="      No changes were saved."    RT
047210      GOTO CWMENU                          RT
047310      RT                                  RT
047410 UPDATE: READ #1, HOLD, KEY=WORD$, IOBUFF$ RT
047510      PROTECT$ = " " :P$=0                  RT
047610      $PACK (P=F1$) IOBUFF$ FROM WORD$,SUBONE$,SUBTWO$, : RT
047710      PROTOSPC$,PPFL$,NEGFL$,PROTECT$      RT
047810      REWRITE #1,IOBUFF$,DATA GOTO IOERR2  RT
047910      ER1$= " "                             RT
048010      IF NF=0 THEN RITE                      RT
048110      IF DF=1 THEN EXITDEL                  RT
048210      ZAP$=" "                               RT
048310 ER$="      "&WORD$&" updated - Please select next word." JM
048410      E$=HEX(8C)                            RT
048510      WORD$=" "                               RT
048610      GOTO CWMENU                          RT
048710 RITE:                                       RT
048810 ER$= "      "&WORD$&" inserted - Please select your next word." RT
048910      ZAP$=" "                               RT
049010      E$= HEX(8C)                            RT
049110      WORD$=" "                               RT
049210      GOTO CWMENU                          RT
049310      RT                                  RT
049410 EXITDEL: DF=0                              RT
049510      ER$= "      Deletion cancelled."        RT
049520      X$=HEX(81)                            JM
049530      PASSWRD$=HEX(000102030E0F101112131E1F20) JM
049610      GOTO CWMENU                          RT
049710      RT                                  RT
049810 DEFIN:                                       RT
049811 ACCEPT                                     :
049812      AT (01,01),                           :
049813 "CRES: Word Substitution List Maint Subsystem", :
049814      AT (01,51),                             : JM
049815 "(Definition of Entries Screen)",           : JM

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 15:05 11/13/84 Page 10  
Input File is CRES0WSL in Library CRES0SRC on Volume CRES02

```

049816      AT (02,01), "=====
049817      =====",
049818      AT (05,08), FAC(HEX(AD)), IFLE5$ , CH(14), JM
049819      AT (05,49), FAC(HEX(AD)), IFLE6$ , CH(10), JM
049820      AT (07,03), FAC(HEX(84)), P$(1) , CH(03),
049821      AT (07,08),
049822      "Adjective .....Modifies a noun or pronoun",
049823      AT (08,03), FAC(HEX(84)), P$(2) , CH(03),
049824      AT (08,08),
049825      "Adjective/verb .....A word that may function as an adjective
049826      or verb",
049827      AT (09,03), FAC(HEX(84)), P$(3) , CH(03),
049828      AT (09,08),
049829      "Adverb .....Modifies a verb, adjective, or adverb",
049830      AT (10,03), FAC(HEX(84)), P$(4) , CH(03),
049831      AT (10,08),
049832      "Adverb/verb .....A word that may function as an adverb or
049833      verb",
049834      AT (11,03), FAC(HEX(84)), P$(5) , CH(03),
049835      AT (11,08),
049836      "Article .....A limiting adjective before a noun (e.g.
049837      a, the)",
049838      AT (12,03), FAC(HEX(84)), P$(6) , CH(03),
049839      AT (12,08),
049840      "Conjunction .....Links phrases or clauses (e.g. or)",
049841      AT (13,03), FAC(HEX(84)), P$(7) , CH(03),
049842      AT (13,08),
049843      "Interjection .....Used for simple exclamations (e.g. Oh!)",
049844      ,
049845      AT (14,03), FAC(HEX(84)), P$(8) , CH(03),
049846      AT (14,08),
049847      "Noun .....Indicates a person, place, or thing",
049848      AT (15,03), FAC(HEX(84)), P$(13) , CH(03),
049849      AT (15,08),
049850      "Noun/adjective .....A word that may function as a noun or ad
049851      jective",
049852      AT (16,03), FAC(HEX(84)), P$(9) , CH(03),
049853      AT (16,08),
049854      "Noun/verb .....Word that may function as a noun or verb",
049855      ,
049856      AT (17,03), FAC(HEX(84)), P$(10) , CH(03),
049857      AT (17,08),
049858      "Preposition .....Expresses a relationship between two ele
049859      ments of ",
049860      AT (18,08),
049861      " a sentence (e.g. in, on, by, to, since) JM
049862      ",
049863      AT (19,03), FAC(HEX(84)), P$(11) , CH(03),
049864      AT (19,08),
049865      "Pronoun .....Word that may be used as a substitute fo
049866      r a noun",
049867      AT (20,08),
049868      " (e.g. I, you, she, what,....)", JM
049869      AT (21,03), FAC(HEX(84)), P$(12) , CH(03),
049870      AT (21,08),
049871      "Verb .....Denotes an action or occurrence",
049872      AT (23,20),

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 15:05 11/13/84 Page 11  
Input File is CRES@WSL in Library CRES@SRC on Volume CRES02

```

049873 "Press",
049874 AT (23,26), FAC(HEX(84)), PF1$ , CH(03),
049875 AT (23,30),
049876 "to return to the previous screen",
056210 KEYS(BIN(1)&BIN(17)), ON (BIN(1)&BIN(17)) GOTO ERONINS,ERONINS
056310 RT
056410 NEW: PRINT PAGE, AT(12,28),HEX(84);"Closing last file" RT
056510 FOR I= 1 TO 50000:NEXT I RT
056610 FILES=" "
056710 CLOSE #1 RT
056810 GOTO AGAIN RT
056910 RT
057010 /* THIS FUNCTION PRINTS OUT ERROR MESSAGES AND */ RT
057110 /* SETS UP FLASHING FAC'S IN APPROPRIATE PLACES */ RT
057210 DEF FN'7(MSG$,B$,C$,D$) RT
057310 PRINT RT
057410 M$=HEX(94) RT
057510 G$=B$ RT
057610 L$=C$ RT
057710 V$=D$ RT
057810 RETURN RT
057910 RT
058010 EXIT: CLOSE #1 RT
058110 OUT: /* RETURNING TO SUBSTITUTION LIST MENU */ JM

```

Technical Report 86-001

THIS PAGE INTENTIONALLY LEFT BLANK.

Print Word Substitute List (CRES@WSP)

CRES@WSP in library CRES@SRC is the BASIC language program to print the Word Substitute List. The object language version is CRES@WSP in library CRES@SAV.

This program first allows the user to specify the Word Substitute List to be printed. Then the program enables the user to print out either the entire Word Substitute List or a specified portion. This printout can be displayed at the terminal or sent to the printer. An optional subtitle of up to two lines can be included for the printout, and the number of lines per page may be specified as some value less than the default value of 50.

A complete program listing follows. Appendix C contains a sample flowchart.

Technical Report 86-001

THIS PAGE INTENTIONALLY LEFT BLANK.

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 14:30 10/31/84 Page 1  
Input File is CRES@WSP in Library CRES@SRC on Volume CRES02

```

000010 /* This is the CRES WORD SUBSTITUTION LIST PRINT PROGRAM */ JM
000015 /* */ RT
000020 /*          AUTHORS: Rose Tank and John Mears */ RT
000025 /*          Date:   October 12, 1983 */ RT
000030 /* */ RT
000035 RT
000040 /* The following code sets up a screen, which asks for the */ RT
000045 /* name of the file containing the word substitution list */ RT
000050 /* to be printed. It also provides the opportunity to change */ RT
000055 /* the library and volume names which are set to the appro- */ RT
000060 /* priate defaults through the "Extract" system routine. */ RT
000065 RT
000070 RT
000075 DIM WORD$24,ERR1$53,CODE$1,MSG$53,F$40,G$1,CM$255,MSG2$40 RT
000080 DIM IOBUFF$70,ERR2$40,FIL$22 RT
000085 DIM FILE$08,LIB$08,RLIB$8,RVOL$6,SLIB$8,SVOL$6 JM
000090 DIM IVOL$06,HEADER$51,L$1,V$1,B$1,C$2,D$8,M$1,MSK$2 RT
000095 DIM SBTITLE$20,SBTITLE$20 RT
000100 DIM EN$ 05 RT
000105 DIM ER$ 79 RT
000110 DIM STARTPH$ 24 RT
000115 DIM ENDPH$ 24 RT
000120 DIM TP$ 08 RT
000125 DIM PF15$ 04 RT
000130 DIM PF16$ 04 RT
000135 RT
000140 EN$="ENTER" RT
000145 PF15$ ="PF15" RT
000150 PF16$="PF16" RT
000155 HEADER$ ="CRES WORD SUBSTITUTION LIST PRINT SUBSYSTEM" RT
000160 FIL$ ="Word Substitution List" RT
000170 F$= "Filename not found on (library/volume). " RT
000175 ERR1$= "I/O ERROR - Please specify WORD SUBSTITUTION list." RT
000180 ERR2$="I/O ERROR - Internal format not matched." RT
000185 CALL "EXTRACT" ADDR("IV",IVOL$,"IL",LIB$,"RL",RLIB$,"RV",RVOL$, : JM
000190 "SL",SLIB$,"SV",SVOL$) JM
000192 FILE$="WRDSBLST" JM
000195 AGAIN: G$=HEX(81) RT
000200 MSG$=" ":L$=HEX(81) RT
000205 V$=HEX(81): M$=HEX(8D) RT
000210 RT
000215 FILMENU: RT
000220 ACCEPT : RT
000225 AT (01,01), : RT
000230 "CRES: Word Substitution List Print Subsystem", : RT
000235 AT (01,57), : RT
000240 "(File Selection Screen)", : RT
000245 AT (02,01), : RT
000250 "===== : RT
000255 =====", : RT
000260 AT (04,10), : RT
000265 "Welcome to the", : RT
000270 AT (04,26), FAC(HEX(85)), : RT
000275 HEADER$, : RT
000280 AT (07,10), : RT
000285 "Please specify the", : RT
000290 AT (07,29), FAC(HEX(84)), : RT

```

# Technical Report 00-001

Wang VS Integrated Editor - Version 6.10.00 14:30 10/31/84 Page 2  
Input File is CRES@WSP in Library CRES@SRC on Volume CRES02

```

000295 FILE$,
000300 AT (07,52),
000305 "you want to print,",
000310 AT (08,19),
000315 "if the default values are not satisfactory.",
000350 AT (13,30),
000355 "File =",
000360 AT (13,43),FAC(G$),FILE$ , CH(08),
000365 RANGE ( "!" , "XXXXXXXXXX"),
000370 AT (14,30),
000375 "Library =",
000380 AT (14,43),FAC(L$), LIB$ , CH(08),
000385 RANGE ( "!" , "XXXXXXXXXX"),
000390 AT (15,30),
000395 "Volume =",
000400 AT (15,43),FAC(V$), IVOL$ , CH(06),
000405 RANGE ( "!" , "XXXXXXXXXX"),
000410 AT (18,20), FAC(M$), MSG$,
000415 AT (20,30),"Press",
000420 AT (20,36),FAC(HEX(84)),
000425 EN$,
000430 AT(20,42),
000435 "to continue",
000440 AT (21,17),"Press",
000445 AT (21,23), FAC(HEX(84)),
000450 PF16$,
000455 AT (21,28),
000460 "to return to the Substitution List Menu",
000465 ON (BIN(16)&BIN(32)) GOTO OUT
000475
000480 /* SET UP THE USER FILE BLOCK */
000485
000490 SELECT #1,"WRDSUB",VARC,INDEXED,RECSIZE=80,KEYPOS=1,KEYLEN=20,
000495 IOERR GOTO IOERR1
000500
000505 SELECT #2,"PRTFILE",PRINTER,RECSIZE=80,IOERR GOTO IOERR3
000510
000515 /* SEE IF THE SPECIFIED FILE EXISTS */
000520
000525 CALL "LIMITS"(FILE$,LIB$,IVOL$,CODE$,MSG$)
000530 IF CODE$="Y" THEN ZZZ
000535 IF CODE$="N" THEN GOSUB'7(F$,HEX(91),HEX(81),HEX(81))
000540 IF CODE$="V" THEN GOSUB'7(STR(MSG$,3),HEX(81),HEX(81),HEX(91))
000545 IF CODE$="L" THEN GOSUB'7(STR(MSG$,3),HEX(81),HEX(91),HEX(81))
000550 IF CODE$="E" THEN GOSUB'7(STR(MSG$,3),HEX(81),HEX(81),HEX(81))
000555 GOTO FILMNU
000560
000570 /* IS THIS REALLY A WORD SUBSTITUTION LIST? */
000575
000580 ZZZ:
000585 I%=0
000590 CALL "READFDR" ADDR(FILE$,LIB$,IVOL$,I%,"RS",R%&,R%)
000595 IF R%<>80 THEN IOERR2
000600
000605 /* IF WE FALL HERE, IT MUST BE TIME TO OPEN FILE */
000610
000615 OPEN NOGETPARM #1,SHARED, FILE= FILE$, LIBRARY = LIB$,

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 14:30 10/31/84 Page 3  
Input File is CRES@WSP in Library CRES@SRC on Volume CRES02

```

000620     VOLUME = IVOL$                                RT
000625                                                                RT
000630 /* This section of code generates the second screen, which asks */ RT
000635 /* for the particular word interval and its corresponding */ RT
000640 /* print operation. The user is given the option of returning */ RT
000645 /* to the previous screen to select a new word file to print, */ RT
000650 /* or of returning to the substitution list menu. */ JM
000655                                                                RT
000660                                                                RT
000665 ER$ = " "                                             RT
000670 CWMENU:                                             RT
000675     WORD$ = " "                                       RT
000680     SBTITL1$ = " "                                   RT
000685 BALONEY:                                             RT
000690     STARTPH$ = "FIRST"                                RT
000695     ENDPH$ = "LAST"                                  RT
000700     TP$ = "TERMINAL"                                  RT
000705     LPP% = 50                                         RT
000710                                                                RT
000715 PRINISCR:                                             RT
000720 ACCEPT /                                             ! RT
000725     AT (01,03),                                     ! RT
000730 "CRES: Word Substitution List Print Subsystem",      ! RT
000735     AT (01,66),                                     ! RT
000740 "(Print Screen)",                                  ! RT
000745     AT (02,03),                                     ! RT
000750 "=====": RT
000755 "=====": RT
000760     AT (04,13),                                     ! RT
000765 "To print a range of lines, fill in the information below.", ! RT
000770     AT (06,02), FAC(HEX(85)),                       ! RT
000775 ER$,                                                 ! RT
000780     AT (08,22),                                     ! RT
000785 "Subtitle of list =",                               ! RT
000790     AT (08,42), FAC(HEX(81)), SBTITL1$, CH(20),      ! RT
000795     AT (09,42), FAC(HEX(81)), SBTITL1$, CH(20),      ! RT
000800     AT (11,22),                                     ! RT
000805 "Starting word =",                                 ! RT
000810     AT (11,42), FAC(HEX(81)), STARTPH$, CH(24),      ! RT
000815     RANGE ( ":", " ", "*****"),                  ! RT
000820     AT (12,22),                                     ! RT
000825 "Ending word =",                                 ! RT
000830     AT (12,42), FAC(HEX(81)), ENDPH$, CH(24),        ! RT
000835     RANGE ( ":", " ", "*****"),                  ! RT
000840     AT (14,22),                                     ! RT
000845 "Lines per page =",                               ! RT
000850     AT (14,42), FAC(HEX(82)), LPP%, PIC(##),          ! RT
000855     AT (16,22),                                     ! RT
000860 "Print at (TERMINAL/PRINTER) =",                  ! RT
000865     AT (16,53), FAC(HEX(81)), TP$, CH(08),          ! RT
000870     AT (20,32),                                     ! RT
000875 "Press",                                             ! RT
000880     AT (20,38), FAC(HEX(85)), EN$, CH(05),          ! RT
000885     AT (20,44),                                     ! RT
000890 "to print",                                         ! RT
000895     AT (21,17),                                     ! RT
000900 "Press",                                             ! RT

```

# Technical Report 8t-001

Wang VS Integrated Editor - Version 6.10.00 14:30 10/31/84 Page 4  
Input File is CRESEWSP in Library CRESE\$SRC on Volume CRE\$02

```

000905      AT (21,23), FAC(HEX(84)), PF15$, CH(04),      : RT
000910      AT (21,28),      : RT
000915 "to select another Word Substitution List",      : RT
000920      AT (22,17),      : RT
000925 "Press",      : RT
000930      AT (22,23), FAC(HEX(84)), PF16$, CH(04),      : RT
000935      AT (22,28),      : RT
000940 "to return to the Substitution List Menu",      : RT
000945 KEYS(BIN(0)&BIN(15)&BIN(16)&BIN(31)&BIN(32)),      : RT
000950 ON (BIN(15)&BIN(31)&BIN(16)&BIN(32)) GOTO NEW,NEW,EXIT,EXIT      : RT
000955      IF STR(TP$,1,1) = "P" THEN NEXT11      : RT
000960      IF STR(TP$,1,1) <> "T" THEN TERMPR      : RT
000965 NEXT11:      IF LPP% < 51 THEN NEXT13      : RT
000970      ER$="Lines per page' can be at most 50."      : RT
000975      GOTO PRINTSCR      : RT
000980 IOERR1: PRINT      : RT
000985 PRINT BELL,HEX(94),ERR1$      : RT
000990 FOR I=1 TO 200000 :NEXT I      : RT
000995 GOTO FILMNU      : RT
001000      : RT
001005 IOERR2: MSG$=ERR1$      : RT
001010 MS$= HEX(84)      : RT
001015 GOTO FILMNU      : RT
001020 IOERR3: PRINT      : RT
001025 PRINT BELL, HEX(94), ERR2$      : RT
001030 FOR I=1 TO 500000 :NEXT I      : RT
001035 CLOSE #2      : RT
001040 GOTO BALONEY      : RT
001045 TERMPR:      : RT
001050 ER$="Print at ' has only legal values TERMINAL or PRINTER."      : RT
001055 GOTO PRINTSCR      : RT
001060      : RT
001065 REX: ER$="      'Starting word' comes after 'Ending word'."      : RT
001070 GOTO PRINTSCR      : RT
001075      : RT
001080      : RT
001085 NEXT13: IF STARTPH$="FIRST" OR ENDPH$="LAST" THEN FORD      : JM
001090 IF STARTPH$>ENDPH$ THEN REX      : JM
001095      : RT
001100 /* Now it's time to generate our print file name */      : RT
001105      : RT
001110 FORD: DAT$= DATE      : JM
001115 DS$= STR(DAT$,3,2)&"/"&STR(DAT$,5,2)&"/"&STR(DAT$,1,2)      : RT
001120 TS$= STR(TIME,1,2)&":"&STR(TIME,3,2)      : RT
001125      : RT
001130 DISPLAY      : JM
001135 AT (01,03),      : RT
001140 "CRES: Word Substitution List Print Subsystem",      : JM
001145 AT (01,61),      : RT
001150 "(Generation Screen)",      : JM
001155 AT (02,03),      : RT
001160 "=====      : RT
001165 =====",      : RT
001170 AT (12,23),      : RT
001175 "Output file now being generated!"      : JM
001180 PF16$="WRDSUB##"      : RT
001185 I%=0      : RT

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 14:30 10/31/84 Page 5  
Input File is CRES@WSP in Library CRES@SRC on Volume CRES02

```

001190 NEXT14: CONVERT I% TO STR(PFILE$,7,2),PIC(00)      RT
001195      I%=I%+1                                         RT
001200      CALL "LIMITS" (PFILE$,SLIB$,SVOL$,CODE$,MSG$)   JM
001205      IF CODE$="Y" THEN NEXT14                        RT
001207      IF CODE$="E" THEN ER$ = STR(MSG$,3)             JM
001209      IF CODE$="E" THEN PRINTSCR                      JM
001210      OPEN NOGETPARM #2,OUTPUT,FILE=PFILE$,LIBRARY=SLIB$, ! JM
001215      VOLUME=SVOL$,SPACE=8000                        JM
001220      GOSUB TTL_PGE                                    RT
001225      IF STARTPH$ = "FIRST" THEN STARTPH$ = ALL(HEX(00)) RT
001230      IF ENDPH$ = "LAST" THEN ENDPH$ = ALL(HEX(FF))   RT
001235      PAGE% = 1                                         RT
001240      RECS% = 1                                         RT
001245      READ #1, KEY>=STARTPH$,IOBUFF$                 RT
001250      GOSUB HEADER                                       RT
001252      IF STR(IOBUFF$,1,18)=" WORD SUBSTITUTION" THEN READ #1, ! JM
001254      IOBUFF$, EOD GOTO HOME                            JM
001255      IF STR(IOBUFF$,20)>STR(ENDPH$,20) THEN HOME      RT
001260      GOTO KNOT                                          RT
001265 AGAN: READ #1, IOBUFF$,EOD GOTO HOME                RT
001270      IF STR(IOBUFF$,20)>STR(ENDPH$,20) THEN HOME      RT
001275      IF RECS% = 1 THEN GOSUB HEADER                  RT
001280 KNOT:                                               RT
001285      WRITE #2, USING CAT6,C$,STR(IOBUFF$,1,20),      ! RT
001290      STR(IOBUFF$,21,20),STR(IOBUFF$,41,15)            RT
001300      RECS%=RECS%+1                                     RT
001305      IF RECS%>LPP% THEN RECS%=1                     RT
001310      IF RECS%=1 THEN PAGE%=PAGE%+1                   RT
001315      GOTO AGAN                                        RT
001320 HOME: WRITE #2, USING DOG, C$                        RT
001325      WRITE #2, USING DOG, C$                          RT
001330      WRITE #2, USING CAT8, C$                          RT
001335      CLOSE#2                                           RT
001340      IF STR(TP$,1,1) ="P" THEN MESS                 RT
001345      CLOSE WS                                           RT
001350      MSG2$ = "CANCEL Processing"                       RT
001355      MSK$=HEX(00)                                       RT
001360      CMSG$ = " "                                         RT
001365      CALL "LINK" ADDR("DISPFILE","P",RLIB$,RVOL$,3%,PFILE$, ! JM
001370      SLIB$,SVOL$,"C",MSG2$,27%,"H",MSK$,CMSG$,255%,C$,R%) JM
001375      ER$ = " "                                           RT
001380      IF C%<>0 OR R%<>0                                  ! RT
001385      THEN ER$ = STR(CMSG$,141)                         RT
001390      GOTO CWMENU                                         RT
001395      RT                                                 RT
001400 MESS: ER$="      File "&PFILE$&" in Library "&SLIB$&" on "& ! JM
001405      SVOL$&" generated."                                JM
001410      GOTO BALONEY                                         RT
001415      RT                                                 RT
001420 NEW: PRINT PAGE, AT(12,28),HEX(84);"Closing last file" RT
001425      FOR I= 1 TO 50000 :NEXT I                        RT
001430      CLOSE #1:FILE$= " "                               RT
001435      GOTO AGAIN                                         RT
001440      RT                                                 RT
001445 TTL_PGE:                                             RT
001450      C$=HEX(8001)                                         RT
001455      WRITE #2, USING DOG, C$                          RT

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 14:30 10/31/84 Page 6  
Input File is CRESEWSP in Library CRESESRC on Volume CRESU2

```

001460      C$ = HEX(0015)                                RT
001465      WRITE #2, USING DOG, C$                        RT
001470      C$ = HEX(0001)                                RT
001475      WRITE #2, USING DOG1, C$                       RT
001480      WRITE #2, USING DOG2, C$                       RT
001485      WRITE #2, USING DOG0, C$,D$,T$                RT
001490      WRITE #2, USING DOG3, C$                       RT
001495      WRITE #2, USING DOG2, C$                       RT
001500      WRITE #2, USING DOG1, C$                       RT
001505      WRITE #2, USING DOG2, C$                       RT
001510      IF SBTITLE$ = "                                " THEN GERRY RT
001515      WRITE #2, USING DOG4, C$, SBTITLE$              RT
001520      IF SBTITL1$<>"                                " THEN WRITE #2, USING : RT
001525      DOG10, C$, SBTITL1$                             RT
001530      WRITE #2, USING DOG2, C$                       RT
001535      GERRY: WRITE #2, USING DOG5, C$, FILE$          RT
001540      WRITE #2, USING DOG2, C$                       RT
001545      WRITE #2, USING DOG6, C$, LIB$                 RT
001550      WRITE #2, USING DOG2, C$                       RT
001555      WRITE #2, USING DOG7, C$, IVOL$               RT
001560      WRITE #2, USING DOG2, C$                       RT
001565      WRITE #2, USING DOG8, C$, STARTPH$            RT
001570      WRITE #2, USING DOG2, C$                       RT
001575      WRITE #2, USING DOG9, C$, ENDPH$              RT
001580      WRITE #2, USING DOG2, C$                       RT
001585      WRITE #2, USING DOG1, C$                       RT
001590      RETURN                                          RT
001595                                                    RT
001600      HEADER:                                       RT
001605      IF PAGE%=1 THEN FIRST                         RT
001610      C$ = HEX(0001)                                RT
001615      WRITE #2, USING DOG, C$                       RT
001620      FIRST: C$ = HEX(8001)                         RT
001625      WRITE #2, USING DOG, C$                       RT
001630      C$ = HEX(0001)                                RT
001635      IF SBTITLE$ = "                                " THEN : RT
001640      THEN WRITE #2, USING CAT,C$,D$,T$,PAGE%         : RT
001645      ELSE GOTO FOUT                                  RT
001650      GOTO ALRITE                                     RT
001655      FOUT: WRITE #2, USING CAT1,C$,SBTITLE$,D$,T$,PAGE% RT
001660      IF SBTITL1$<>"                                " THEN : RT
001665      THEN WRITE #2, USING CAT7,C$,SBTITL1$          RT
001670      ALRITE: WRITE #2, USING CAT2, C$, FILE$, LIB$, IVOL$ RT
001675      WRITE #2, USING CAT3, C$                       RT
001680      WRITE #2, USING CAT4, C$                       RT
001685      WRITE #2, USING CAT5, C$                       RT
001690      WRITE #2, USING CAT3, C$                       RT
001695      RETURN                                          RT
001700                                                    RT
001705      DOG0:### : ##### : RT
001710      ##### : RT
001715      DOG: ### RT
001720      DOG1:### :-----! RT
001725      -----! : RT
001730      DOG2:### : RT
001735      : RT
001740      DOG3:### : ***** WORD SUBSTITUTION LIST : RT

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 14:30 10/31/84 Page 7  
Input File is CRESEWSP in Library CRES@SRC on Volume CRES02

```

001745 ***** ! RT
001750 DOG4:### ! TITLE OF LIST: #####! RT
001755 ##### ! RT
001760 DOG5:### ! INPUT FILE: ##### ! RT
001765 ! RT
001770 DOG6:### ! INPUT LIBRARY: ##### ! RT
001775 ! RT
001780 DOG7:### ! INPUT VOLUME: ##### ! RT
001785 ! RT
001790 DOG8:### ! STARTING WORD: #####! RT
001795 ##### ! RT
001800 DOG9:### ! ENDING WORD: #####! RT
001805 ##### ! RT
001810 DOG10:### ! #####! RT
001815 ##### ! RT
001820 CAT:### ##### ! RT
001825 PAGE ##### RT
001830 CAT1:### TITLE OF LIST : ##### ! RT
001835 ##### PAGE ##### RT
001840 CAT7:### ##### RT
001845 CAT2:### Input File is ##### in Library ##### on Volume ###! RT
001850 ### RT
001855 CAT3:###===== ! RT
001860 ===== RT
001870 *CAT4:### WORD First Second RT
001875 * Part of Past RT
001885 *CAT5:### Suostitute Suostitute RT
001890 * Speech Part. RT
001900 *CAT6:### ##### RT
001905 * # RT
001910 CAT4:### WORD First Second RT
001915 RT
001920 CAT5:### Substitute Substitute RT
001925 RT
001930 CAT6:### #####! RT
001935 # RT
001940 CAT8:### !!! END OF LISTING !!! RT
001945 RT
001950 /* THIS FUNCTION PRINTS OUT ERROR MESSAGES AND */ RT
001955 /* SETS UP FLASHING FAC'S IN APPROPRIATE PLACES */ RT
001960 DEF FN'7(MSG$,B$,C$,D$) RT
001965 PRINT RT
001970 M$=HEX(94) RT
001975 G$=B$ RT
001980 L$=C$ RT
001985 V$=D$ RT
001990 RETURN RT
001995 RT
002000 EXIT: CLOSE #1 RT
002005 OUT: /* RETURNING TO SUBSTITUTION LIST MENU */ JM

```

Technical Report 86-001

THIS PAGE INTENTIONALLY LEFT BLANK.

Create/Maintain Phrase Substitute List (CRES@PSL)

CRES@PSL in library CRES@SRC is the BASIC language program to create/maintain the Phrase Substitute List. The object language version is CRES@PSL in library CRES@SAV.

This program first allows the user to specify the Phrase Substitute List to be created or maintained. Then, the program enables the user to add, delete, or modify any given phrase and its substitute(s). Up to two substitutes can be specified of up to 20 characters each.

The only characters accepted in the input phrase or its substitute(s) are the letters A through Z, ".", "-", and "'". There can be no embedded blanks. A complete data record description is provided in appendix A.

There is a protection system implemented for this list, and the program makes provision for clearing the protect bytes, if for some reason the bytes have been left set incorrectly.

A complete program listing follows. Appendix C contains a sample flowchart.

Technical Report 86-001

THIS PAGE INTENTIONALLY LEFT BLANK.

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 13:19 11/09/84 Page 1  
Input File is CRES@PSL in Library CRES@SRC on Volume CRES02

```

000010 /* This is the CRES PHRASE SUBSTITUTION MAINTENANCE SUBSYSTEM */ RT
000020 /* */ RT
000030 /* AUTHORS: Rose Tank and John Mears */ RT
000040 /* Date: October 12, 1983 */ RT
000050 /* */ RT
000060 RT
000070 /* The following code sets up a screen, which asks for the */ RT
000080 /* name of the file containing the Phrase Sub. list that is */ RT
000090 /* to be modified. It also provides the opportunity to change */ RT
000100 /* the library and volume names which are set to the appro- */ RT
000110 /* priate defaults through the "Extract" system routine. */ RT
000120 RT
000130 RT
000140 DIM WORD$24,ERR1$50,CODE$1,MSG$50,F$40,G$1,PROTECT$3 RT
000150 DIM F1$20,IOBUFF$70,ERR1$50,XYZ$1,PASSWRD$13,X$1 JM
000160 DIM PFK$4,TAB$3,BTAB$7,FILE$8,USER$3,USERNAME$24,WRKST$3 RT
000170 DIM LIB$8,FIL$24,MESING$16 JM
000180 DIM IVOL$6,HEADER$50,L$1,V$1,A$1,C$1,D$1,M$1 JM
000190 DIM MOD$15,SUBONE$20,SUBTWO$20 JM
000200 DIM DEF$19, JM
000210 DIM LEG$5 JM
000220 DIM ENTER$5,NOEXIST$68,Z1$1 RT
000230 DIM EX$4 JM
000240 DIM ER$68 JM
000250 DIM ZAP$25 JM
000260 DF=0 JM
000270 ENTER$="ENTER" RT
000280 TAB$="TAB" JM
000290 BTAB$="BACKTAB" JM
000300 SELECT #1,"PHRSUB",VARC,INDEXED,RECSIZE=70,KEYPOS=1,KEYLEN=24, ! JM
000310 IOERR GOTO IOERR1 JM
000320 F1$=HEX(A018A014A014A0030003) JM
000330 F$= "Filename not found on (library/volume)." JM
000340 ERR1$= "I/O ERROR - Please check filename and respecify." JM
000350 CALL "EXTRACT" ADDR("ID",USER$,"IL",LIB$,"IV",IVOL$) JM
000360 FILE$="PHRSBLST" JM
000370 AGAIN: G$=HEX(81) JM
000380 MSG$=" ":L$=HEX(81) JM
000390 V$=HEX(81): M$=HEX(8D) JM
000400 HEADER$ ="CRES PHRASE SUBSTITUTION MAINTENANCE SUBSYSTEM" JM
000410 FIL$ ="Phrase Substitution List" RT
000420 EX$="PF16" JM
000430 JM
000440 FILMNU: JM
000450 ACCEPT ! JM
000460 AT (01,01), ! JM
000470 "CRES: Phrase Substitution List Maintenance Subsystem", ! JM
000480 AT (01,57), ! JM
000490 "(File Selection Screen)", ! JM
000500 AT (02,01), ! RT
000510 "===== RT
000520 =====", ! RT
000530 AT (04,10), ! RT
000540 "welcome to the", ! RT
000550 AT (04,25), FAC(HEX(85)), ! RT
000560 HEADER$, ! RT
000570 AT (07,04), ! RT

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 13:19 11/09/84 Page 2  
Input File is CRES@PSL in Library CRES@SRC on Volume CRES02

```

000580 "Please specify the",
000590 AT (07,23), FAC(HEX(84)),
000600 FILE$,
000610 AT (07,48),
000620 "you want to modify or create,",
000630 AT (08,04),
000640 "if the default values are not satisfactory. ",
000650 AT (13,30),
000660 "File =",
000670 AT (13,43),FAC(G$),FILE$, CH(08),
000680 RANGE ( ":", " ", "XXXXXXXXXX"),
000690 AT (14,30),
000700 "Library =",
000710 AT (14,43),FAC(L$), LIB$, CH(08),
000720 RANGE ( ":", " ", "XXXXXXXXXX"),
000730 AT (15,30),
000740 "Volume =",
000750 AT (15,43),FAC(V$), IVOL$, CH(06),
000760 RANGE ( ":", " ", "XXXXXXXXXX"),
000770 AT (18,24), FAC(M$), MSG$,
000780 AT (20,30),"Press",
000790 AT (20,36),FAC(HEX(84)),
000800 ENTER$,
000810 AT (20,42),
000820 "to continue",
000830 AT (21,15),"Press",
000840 AT (21,21), FAC(HEX(84)),
000850 EX$ ,
000860 AT (21,26),
000870 "to return to the Substitution List Menu",
000880 ON (BIN(16)&BIN(32)) GOTO OUT,OUT
000930
000940 /* SEE IF THE SPECIFIED FILE EXISTS */
000950
000960 CALL "LIMITS"(FILE$,LIB$,IVOL$,CODE$,MSG$)
000970 IF CODE$="Y" THEN GOTO ZZZ
000980 IF CODE$="N" THEN CREATE
000990 IF CODE$="V" THEN GOSUB'7(STR(MSG$,3),HEX(81),HEX(81),HEX(91))
001000 IF CODE$="L" THEN GOSUB'7(STR(MSG$,3),HEX(81),HEX(91),HEX(81))
001010 IF CODE$="E" THEN GOSUB'7(STR(MSG$,3),HEX(81),HEX(81),HEX(81))
001020 GOTO FILEMENU
001030
001040 /* IF WE FALL HERE, IT MUST BE TIME TO OPEN FILE */
001050 CREATE:
001060 DIM PF1$ 03
001070 PF1$ ="PF1"
001080 ACCEPT
001090 AT(01,01),
001100 "CRES: Phrase Substitution List Maintenance Subsystem",
001110 AT (01,60),
001120 "(Create File Screen)",
001130 AT (02,01),
001140 "=====
001150 =====",
001160 AT (06,08),
001170 "The file you have selected does not currently exist in the libra
001180 ry",

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 13:19 11/09/84 Page 3  
Input File is CRES@PSL in Library CRES@SRC on Volume CRES02

```

001190      AT (07,08),                                : RT
001200 "and volume specified. For your convenience, the following are ", : JM
001210      AT (08,08),                                : RT
001220 "the parameters you submitted.",                : RT
001230      AT (11,30),                                : RT
001240 "File      =",                                : RT
001250      AT (11,40),FAC(HEX(85)),FILE$                , CH(08), : RT
001260      AT (12,30),                                : RT
001270 "Library=",                                : RT
001280      AT (12,40), FAC(HEX(85)), LIB$                , CH(08), : RT
001290      AT (13,30),                                : RT
001300 "Volume =",                                : RT
001310      AT (13,40), FAC(HEX(85)), IVOL$                , CH(06), : RT
001320      AT (17,15),                                : RT
001330 "To create a new file from these parameters, press", : JM
001340      AT (17,65), FAC(HEX(85)), ENTER$                , CH(05), : RT
001350      AT (19,26),                                : RT
001360 "To select another file, press",                : JM
001370      AT (19,56), FAC(HEX(85)), PF1$                , CH(03), : RT
001380 KEYS(BIN(0)&BIN(1)&BIN(17)),KEY(OPTION),          : RT
001390 ON (BIN(1)&BIN(17)) GOTO FILMNU,FILMNU            : RT
001400                                                    : RT
001410      OPEN NOGETPARM #1, OUTPUT, FILE= FILE$, LIBRARY=LIB$, : JM
001420      VOLUME= IVOL$, SPACE=1000                    : JM
001430      IOBUFF$= " PHRASE SUBSTITUTION"              : JM
001440      WRITE #1, IOBUFF$                             : JM
001450      CLOSE #1                                       : JM
001460      GOTO ZZ1                                       : JM
001470 ZZZ: CALL "READFDR" ADDR(FILE$,LIB$,IVOL$,0%,"RS",RSIZ%,REF%) : JM
001480      IF RSIZ%<>70 THEN ERROR                        : JM
001490 ZZ1: OPEN NOGETPARM #1,SHARED, FILE= FILE$, LIBRARY = LIB$, : JM
001500      VOLUME = IVOL$                                : RT
001510      READ #1, IOBUFF$                             : JM
001520      IF STR(IOBUFF$,1,20) =" PHRASE SUBSTITUTION" THEN YES : JM
001530      CLOSE #1                                       : JM
001540 ERROR: M$=HEX(94)                                : JM
001550      MSG$= ERR1$                                     : JM
001560      GOTO FILMNU                                    : JM
001570                                                    : RT
001580 YES:                                              : RT
001590 /* This section of code generates another screen, which asks */ : JM
001600 /* for the particular word and its corresponding operation - */ : JM
001610 /* either insert word, delete word, or modify word. The user is */ : RT
001620 /* also at this point given the option of selecting another */ : JM
001630 /* phrase sub. list or of returning to the Substitution List */ : JM
001631 /* Menu.                                           */ : JM
001640                                                    : RT
001650 ZAP$="Please select the phrase:"                    : RT
001660 WORD$=" "                                           : RT
001670 WORDEQ$="phrase = "                                : RT
001680 PPK$=" PF "                                         : RT
001690 OPER$=" Operation"                                : RT
001700 OS$="1"                                           : RT
001710 TS$="2":TH$="3":QS$="14":RS$="15":SS$="16"       : RT
001720 ER$ = " "                                           : RT
001730 ER1$=" "                                           : RT
001740 PASSWRD$=HEX(000102030E0F101112131E1F20)        : JM

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 13:19 11/09/84 Page 4  
Input File is CRESEPSL in Library CRESESRC on Volume CRES02

```

001750 X$=HEX(81) JM
001760 NOEXISTS$="The phrase you are attempting to modify does not exist"! RT
001770      &" in this list." RT
001780 Z1$=HEX(84) RT
001790 RT
001800 CWMENU: RT
001820 ACCEPT : RT
001830      AT (01,01), : JM
001840      "CRES: Phrase Substitution List Maintenance Subsystem", : JM
001850      AT (01,56), "(Phrase & Option Screen)", : JM
001860      AT (02,01), "===== RT
001870      =====", : RT
001880      AT (04,11), FAC(HEX(8C)), : RT
001890 ZAP$, : RT
001900      AT (06,27), FAC(HEX(8C)), : RT
001910 WORDEQ$, : RT
001920      AT (06,36), FAC(X$), WORD$, , CH(24), : JM
001930      AT (08,10), FAC(Z1$), ER$, : RT
001940      AT (09,18), FAC(Z1$),ER1$, : RT
001950      AT (11,08), : JM
001960 "To continue, select the PF key corresponding to the desired oper: JM
001970 ation.", : JM
001980      AT (13,30), FAC(HEX(A5)), : RT
001990 PFK$, : RT
002000      AT (13,38), FAC(HEX(AC)), : RT
002010 OPER$, : RT
002020      AT (15,31), FAC(HEX(84)), : RT
002030 O$, : RT
002040      AT (15,39), : RT
002050 "INSERT phrase", : JM
002060      AT (16,31), FAC(HEX(84)), : RT
002070 I$, : RT
002080      AT (16,39), : RT
002090 "DELETE phrase", : JM
002100      AT (17,31), FAC(HEX(84)), : RT
002110 TH$, : RT
002120      AT (17,39), : RT
002130 "MODIFY phrase", : JM
002140      AT (19,31), FAC(HEX(84)), : RT
002150 J$, : RT
002160      AT (19,39), : RT
002170 "Protection Override", : RT
002180      AT (20,31), FAC(HEX(84)), : RT
002190 R$, : RT
002200      AT (20,39), : RT
002210 "Return to Phrase Sub List Selection", : JM
002220      AT (21,31), FAC(HEX(84)), : RT
002230 S$, : RT
002240      AT (21,39), : RT
002250 "Return to Substitution List Menu", : RT
002260 KEYS(PASSWRD$), : JM
002270      KEY(OPTION) : JM
002280 IF OPTION=15 OR OPTION=31 THEN NEW : JM
002290 IF OPTION=16 OR OPTION=32 THEN EXIT : JM
002300 : RT
002310 FOR I=1 TO LEN(WORD$)
002320 IF POS("ABCDEFGHIJKLMNPQRSTUVWXYZ- " STR(WORD$,I,1)) = 0 :

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 13:19 11/09/84 Page 5  
Input File is CRES@PSL in Library CRES@SRC on Volume CRES02

```

002330                                THEN ILLERR
002340 NEXT I
002350
002360 Z1$= HEX(8D)                                RT
002370 ER1$,ER$ = " "                                JM
002380 IF (OPTION=0)AND(DF=1) THEN DEL                RT
002390 IF DF=1 THEN UPDATE                            RT
002400 IF (OPTION=14) OR (OPTION=30) THEN OVERRIDE    RT
002410 IF STR(WORD$,1,1)<>" " THEN GOTO REED         RT
002420                                              RT
002430 PRINT BELL                                     RT
002440 ER$="                                The first character cannot be blank."
002450 ER1$= "Please reenter the appropriate phrase and PF key." RT
002460 Z1$= HEX(94)                                RT
002470 GOTO CWMENU                                  RT
002480                                              RT
002490 ILLERR:
002500 ER$="Illegal character found in word - please reenter."
002510 Z1$=HEX(94): GOTO CWMENU
002520
002530 OVERRIDE:                                RT
002540 IF NF=1 AND P%=1 THEN CHK                    RT
002550 ER$= "          First check if phrase exists by selecting an option." RT
002560 Z1$=HEX(84)                                RT
002570 GOTO CWMENU                                  RT
002580                                              RT
002590 CHK:                                        RT
002600 IF USER$=PROTECT$ THEN REMOVE                RT
002610 ER$= "You don't have the clearance to override the protect." RT
002620 ER1$=" "                                      RT
002630 Z1$= HEX(84)                                RT
002640 GOTO CWMENU                                  RT
002650                                              RT
002660 REMOVE: PROTECT$= " " :P%=0                 RT
002670 READ #1, HOLD, KEY=WORD$,IOBUFF$            RT
002680 $PACK (F=F1$) IOBUFF$ FROM WORD$, SUBONE$,SUBTWO$, ! RT
002690 PROTECT$                                     RT
002700 REWRITE #1, IOBUFF$, DATA GOTO IOERR2        RT
002710 ER$= "          You may now select your option." RT
002720 ER1$=" "                                      RT
002730 Z1$= HEX(84)                                RT
002740 GOTO CWMENU                                  RT
002750                                              RT
002760 IOERR1: PRINT                                RT
002770 M$=HEX(94)                                    RT
002780 MSG$= ERR1$                                    RT
002790 CLOSE #1                                      RT
002800 GOTO FILMNU                                    RT
002810                                              RT
002820 IOERR2: PRINT                                RT
002830 PRINT HEX(94),"Error occurred on file "; FILE$ RT
002840 PRINT HEX(94),"Error code = "; FS(#1)         RT
002850 STOP "Touch ENTER to return to menu."         RT
002860 GOTO CWMENU                                    RT
002870                                              RT
002880 REED: IF OPTION<>0 THEN SRCH                  RT
002890 ER$= "          Command missing."             RT

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 13:19 11/09/84 Page 6  
Input File is CRES@PSL in Library CRES@SRC on Volume CRES02

```

002900 Z1$= HEX(94) RT
002910 GOTO CWMENU RT
002920 RT
002930 SRCH: P%=1 RT
002940 READ #1, HOLD, KEY=WORD$, IOBUFF$, EOD GOTO NOTFOUND, ! RT
002950 DATA GOTO IOERR2 RT
002960 RT
002970 NF=1 RT
002980 RT
002990 /* Unpack and check if currently being modified by another */ RT
003000 RT
003010 $UNPACK (F=F1$) IOBUFF$ TO WORD$, SUBONE$, SUBTWO$, ! RT
003020 PROTECT$ RT
003030 IF PROTECT$= " " THEN NOW RT
003040 CALL "FINDUSER" ADDR( PROTECT$, USERNAME$, WRKST$) RT
003050 CONVERT WRKST$ TO WRKST$, PIC(###) RT
003060 Z1$= HEX(94) RT
003070 ER$=" Sorry, "& USERNAME$ & " is currently editing " RT
003080 ER1$=" "& WORD$ & "" & " at workstation "& WRKST$ RT
003090 REWRITE #1, IOBUFF$ RT
003100 GOTO CWMENU RT
003110 RT
003120 NOW: PROTECT$= USER$ RT
003130 $PACK (F=F1$) IOBUFF$ FROM WORD$, SUBONE$, SUBTWO$, ! RT
003140 PROTECT$ RT
003150 REWRITE #1, IOBUFF$, DATA GOTO IOERR2 RT
003160 RT
003170 /* NOW BRANCH TO THE OPTION SELECTED */ RT
003180 OPT: ON OPTION GOTO INSERT, DELET, MODIFY RT
003190 ON OPTION-16 GOTO INSERT, DELET, MODIFY RT
003200 RT
003210 NOTFOUND: NF=0 RT
003220 GOTO OPT RT
003230 RT
003240 INSERT: RT
003250 /* NF=0 indicates the word doesn't yet exist so go ahead */ RT
003260 /* and insert. NF=1 indicates the word exists. */ RT
003270 RT
003280 IF NF=1 THEN EXIST RT
003290 MOD$= "(Insert Screen)" RT
003291 MESING$="inserting phrase" JM
003300 SUBONE$, SUBTWO$=" " RT
003310 PROTECT$= USER$ RT
003320 $PACK (F=F1$) IOBUFF$ FROM WORD$, SUBONE$, SUBTWO$, ! RT
003330 PROTECT$ RT
003340 WRITE #1, IOBUFF$, DATA GOTO IOERR2 RT
003350 GOTO INS RT
003390 RT
003400 EXIST: RT
003410 ER$=" The phrase "& WORD$ & " already exists in this list." RT
003420 Z1$= HEX(94) RT
003430 READ #1, HOLD, KEY= WORD$, IOBUFF$, DATA GOTO IOERR1 RT
003440 PROTECT$= " ": P%=0 RT
003450 $PACK (F=F1$) IOBUFF$ FROM WORD$, SUBONE$, SUBTWO$, ! RT
003460 PROTECT$ RT
003470 REWRITE #1, IOBUFF$, DATA GOTO IOERR2 RT
003480 GOTO CWMENU RT

```

Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 13:19 11/09/84 Page 7  
Input File is CRES@PSL in Library CRES@SRC on Volume CRES02

```

003490                                RT
003500 DELETE: IF NF=0 THEN MODEL      RT
003510                               DF=1 RT
003520                               ER$= "        Press ENTER to confirm deletion of "& WORD$ RT
003530                               ERI$= "          or PF14 to cancel the deletion." RT
003540                               ZI$= HEX(94) RT
003550                               PASSWRD$=HEX(000E1E) JM
003560                               X$=HFX(84) JM
003570                               GOTO CWMENU RT
003580                                RT
003590 DEL:   READ #1, HOLD,KEY=WORD$,IOBUFF$ RT
003600                               DELETE #1 RT
003610                               DF=0 RT
003620                               ER$= "    '&WORD$&' deleted - Please select your next phrase." RT
003630                               ZAP$=" " RT
003631                               NF=0 RT
003640                               X$=HEX(81) JM
003650                               PASSWRD$=HEX(000102030E0F101112131E1F20) JM
003660                               ZI$= HEX(8C) RT
003670                               WORD$=" " RT
003680                               GOTO CWMENU RT
003690 MODEL:                                RT
003700     ER$= "        The phrase '&WORD$&' does not exist in this list." RT
003710     ZI$= HEX(94) RT
003720     GOTO CWMENU RT
003730                                RT
003740 MODIFY: IF NF=1 THEN GETIT RT
003750     ER$ = NOEXIST$ /* if word not in list, */ RT
003760     ZI$=HEX(94) /* set up message and fac and */ RT
003770     GOTO CWMENU /* go back up to previous screen */ RT
003780                                RT
003790 GETIT:/*READ #1,HOLD,KEY=WORD$,IOBUFF$,EOD GOTO NOTFOUND, */ RT
003800 /*DATA GOTO IOERR2 */ RT
003810                                RT
003820                                RT
003830 MOD$              =(Modify Screen)" RT
003831 MESING$="modifying phrase" JM
003840 INS:                                JM
003850 DEF$              ="CURRENT SUBSTITUTES" JM
003860 LEG$              ="LEGAL ENTRIES" RT
003870 ERI$              =" " RT
003880 XYZ$              =HEX(8C) RT
003890 ERONINS: RT
003900                                RT
003910 ACCEPT                                : RT
003920         AT (01,01),                                : RT
003930 "CRES: Phrase Substitution List Maintenance Subsystem", : RT
003940         AT (01,66), FAC(HEX(8C)),MOD$,            : RT
003950         AT (02,01), "=====": RT
003960 =====", : RT
003970         AT (05,07), : RT
003980 "Please enter the appropriate substitutes.", : JM
003990         AT (07,10),FAC(XYZ$),ERI$, : RT
004000         AT (10,33),FAC(HEX(AD)), DEF$ , : RT
004010         AT (12,07), : RT
004020 "PHRASE:", : RT
004030         AT (12,15),FAC(HEX(84)), WORD$, : RT

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 13:19 11/09/84 Page 8  
Input File is CRES@PSL in Library CRES@SRC on Volume CRES02

```

004040      RANGE ( "!" " , "XXXXXXXXXX"),      : RT
004050      AT (14,07),      : RT
004060 "FIRST SUBSTITUTE:",      : RT
004070      AT (14,32), SUBONE$,CH(20),      : RT
004080      AT (16,07),      : RT
004090 "SECOND SUBSTITUTE:",      : RT
004100      AT (16,32), SUBTWO$,CH(20),      : RT
004110      AT (20,17),"Use",FAC(HEX(85)),TAB$,"or",FAC(HEX(85)),      : RT
004120      BTAB$,"to move between modifiable fields.",      : RT
004130      AT (23,30),      : RT
004140 "Press",      : RT
004150      AT (23,36),FAC(HEX(84)),ENTER$,      : RT
004160      AT (23,42),      : RT
004170 "to continue",      : RT
004180      AT (24,20),      : RT
004190 "Press" ,AT(24,26),FAC(HEX(84)),EX$,AT(24,31),      : RT
004200 "to exit without ",      : JM
004201      AT (24,47), FAC(HEX(8C)), MESING$,      : JM
004210 KEYS(BIN(0)&BIN(16)&BIN(32)),KEY(OPTION),      : RT
004220 ON (BIN(16)&BIN(32)) GOTO EXMOD,EXMOD      : RT
004230      : RT
004240 IF SUBONE$<>" " THEN UPDATE      : RT
004250 ER1$=" The First Substitute can not be blank."      : RT
004260 XYZ$= HEX(94)      : RT
004270 GOTO ERONINS      : RT
004280      : RT
004290 EXMOD: IF NF=1 THEN NOCHAN      : RT
004300      READ #1, HOLD,KEY=WORD$,IOBUFF$      : RT
004310      DELETE #1      : RT
004320      ER$=" Phrase "&WORD$&" was not inserted."      : RT
004330      GOTO CWMENU      : RT
004340      : RT
004350 NOCHAN: READ #1, HOLD, KEY=WORD$, IOBUFF$      : RT
004360      PROTECT$ = " " :P%=0      : RT
004370      $PACK (F=F1$) IOBUFF$ FROM WORD$,SUBONE$,SUBTWO$,      : RT
004380      PROTECT$      : RT
004390      REWRITE #1,IOBUFF$,DATA GOTO IOERR2      : RT
004400      ER$=" No changes were saved."      : RT
004410      GOTO CWMENU      : RT
004420      : RT
004430 UPDATE: READ #1, HOLD, KEY=WORD$, IOBUFF$      : RT
004440      PROTECT$ = " " :P%=0      : RT
004450      $PACK (F=F1$) IOBUFF$ FROM WORD$,SUBONE$,SUBTWO$,      : RT
004460      PROTECT$      : RT
004470      REWRITE #1,IOBUFF$,DATA GOTO IOERR2      : RT
004480      ER1$= " "      : RT
004490      IF NF=0 THEN RITE      : RT
004500      IF DP=1 THEN EXDEL      : RT
004510      ZAP$=" "      : RT
004520 ER$=" "&WORD$&" updated - Please select next phrase"      : JM
004530      Z1$=HEX(8C)      : RT
004540      WORD$=" "      : RT
004550      GOTO CWMENU      : RT
004560 RITE:      : RT
004570 ER$=" "&WORD$&" inserted - Please select your next phrase."      : RT
004580      ZAP$=" "      : RT
004590      Z1$= HEX(8C)      : RT

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 13:19 11/09/84 Page 9  
Input File is CRES@PSL in Library CRES@SRC on Volume CRES02

004600	WORDS=" "	RT
004610	GOTO CWMENU	RT
004620		RT
004630	EXDEL: DF=0	RT
004640	ER\$= " Deletion cancelled."	RT
004641	X\$=HEX(81)	JM
004646	PASSWRD\$=HEX(000102030E0F101112131E1F20)	JM
004650	GOTO CWMENU	RT
004660		RT
004670	NEW: PRINT PAGE, AT(12,28),HEX(84);"Closing last file"	RT
004680	FOR I= 1 TO 50000:NEXT I	RT
004690	FILES=" "	
004700	CLOSE #1	RT
004710	GOTO AGAIN	RT
004720		RT
004730	/* THIS FUNCTION PRINTS OUT ERROR MESSAGES AND */	RT
004740	/* SETS UP FLASHING FAC'S IN APPROPRIATE PLACES */	RT
004750	DEF FN'7(MSG\$,B\$,C\$,D\$)	RT
004760	PRINT	RT
004770	M\$=HEX(94)	RT
004780	G\$=B\$	RT
004790	L\$=C\$	RT
004800	V\$=D\$	RT
004810	RETURN	RT
004820		RT
004830	EXIT: CLOSE #1	RT
004840	OUT: /* RETURNING TO SUBSTITUTION LIST MENU */	JM

Technical Report 86-001

THIS PAGE INTENTIONALLY LEFT BLANK.

Print Phrase Substitute List (CRES@PSP)

CRES@PSP in library CRES@SRC is the BASIC language program to print the Phrase Substitute List. The object language version is CRES@PSP in library CRES@SAV.

This program first allows the user to specify the Phrase Substitute List to be printed. Then, the program enables the user to print out either the entire Phrase Substitute List, or some specified portion thereof. This printout can be displayed at the terminal or sent to the printer. An optional subtitle of up to two lines can be included for the printout, and the number of lines per page may be specified as some value less than the default value of 50.

A complete program listing follows. Appendix C contains a sample flowchart.

Technical Report 86-001

THIS PAGE INTENTIONALLY LEFT BLANK.

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 16:17 11/13/84 Page 1  
Input File is CRES@PSP in Library CRES@SRC on Volume CRES02

```

000010 /* This is the CRES PHRASE SUBSTITUTION LIST PRINT PROGRAM */ RT
000020 /* */ RT
000030 /* AUTHORS: Rose Tank and John Mears */ RT
000040 /* Date: October 12, 1983 */ RT
000050 /* */ RT
000060 RT
000070 /* The following code sets up a screen, which asks for the */ RT
000080 /* name of the file containing the phrase substitution list */ RT
000090 /* to be printed. It also provides the opportunity to change */ RT
000100 /* the library and volume names which are set to the appro- */ RT
000110 /* priate defaults through the "Extract" system routine. */ RT
000120 RT
000130 RT
000140 DIM WORD$24,ERR1$53,CODE$1,MSG$53,F$40,G$1,CMSG$255,MSG2$40 RT
000150 DIM IOBUFF$70,EXP$31,ERR2$40,RVOL$06,RLIB$8,SLIB$8,SVOL$6 JM
000160 DIM TAB$3,BTAB$7,WORDEQ$9,FILE$08,LIB$08,FIL$24 RT
000170 DIM IVOL$06,HEADER$51,L$1,V$1,B$1,C$2,D$8,M$1,MSK$2 RT
000180 DIM SBTITLE$20,SBTITL1$20 RT
000190 DIM EN$ 05 RT
000200 DIM EX$ 04 RT
000210 DIM ER$ 79,MESSG$21 RT
000220 DIM STARTPH$ 24 RT
000230 DIM ENDPH$ 24 RT
000240 DIM TP$ 08 RT
000250 DIM PF15$ 04 RT
000260 DIM PF16$ 04 RT
000270 RT
000280 EN$ ="ENTER" JEM
000290 TAB$="TAB" RT
000300 BTAB$="BACKTAB" RT
000310 EXP$="(Explanation of Entries Screen)" RT
000320 MESSG$="Parts of Speech Codes" RT
000330 F$="Filename not found on (library/volume)." RT
000340 ERR1$="I/O ERROR - Please specify PHRASE SUBSTITUTION list." RT
000350 ERR2$="I/O ERROR - Internal format not matched." RT
000360 CALL "EXTRACT" ADDR("IL",LIB$,"RV",RVOL$,"RL",RLIB$,"SL",SLIB$, : JM
000361 "SV",SVOL$,"IV",IVOL$) JM
000370 FILE$="PHRSBLST"
000390 AGAIN: G$=HEX(81) RT
000400 MSG$=" ":L$=HEX(81) RT
000410 V$=HEX(81): M$=HEX(8D) RT
000420 HEADER$ ="CRES PHRASE SUBSTITUTION LIST PRINT SUBSYSTEM" RT
000430 FIL$ ="Phrase Substitution List" RT
000440 EX$="PF16" RT
000450 RT
000460 FILMNU: RT
000470 ACCEPT : RT
000480 AT (01,01), : RT
000490 "CRES: Phrase Substitution List Print Subsystem", : RT
000500 AT (01,57), : RT
000510 "(File Selection Screen)", : RT
000520 AT (02,01), : RT
000530 "===== : RT
000540 =====", : RT
000550 AT (04,10), : RT
000560 "Welcome to the", : RT
000570 AT (04,26), PAC(HEX(85)), : RT

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 16:17 11/13/84 Page 2  
Input File is CRES@PSP in Library CRES@SRC on Volume CRES02

```

000580 HEADERS$,
000590 AT (07,10),
000600 "Please specify the",
000610 AT (07,29), FAC(HEX(84)),
000620 FIL$,
000630 AT (07,54),
000640 "you want to print,",
000650 AT (08,19),
000660 "if the default values are not satisfactory. ",
000710 AT (13,30),
000720 "File =",
000730 AT (13,43),FAC(G$),FILE$, CH(08),
000740 RANGE ( ":", " ", "cccccccccccc"),
000750 AT (14,30),
000760 "Library =",
000770 AT (14,43),FAC(L$), LIB$, CH(08),
000780 RANGE ( ":", " ", "cccccccccccc"),
000790 AT (15,30),
000800 "Volume =",
000810 AT (15,43),FAC(V$), IVOL$, CH(06),
000820 RANGE ( ":", " ", "cccccccccccc"),
000830 AT (18,20), FAC(M$), MSG$,
000832 AT (20,30),"Press",
000834 AT (20,36),FAC(HEX(84)),
000836 EN$,
000838 AT(20,42),
000840 "to continue",
000842 AT (21,18),"Press",
000844 AT (21,24), FAC(HEX(84)),
000846 EX$,
000848 AT (21,29),
000870 "to return to the Substitution List Menu",
000880 ON (BIN(16)&BIN(32)) GOTO OUT
000890
000900
000910 /* SET UP THE USER FILE BLOCK */
000920
000930 SELECT #1,"PHRASE",VARC,INDEXED,RECSIZE=70,KEYPOS=1,KEYLEN=24,
000940 IOERR GOTO IOERR1
000950
000960 SELECT #2,"PRFILE",PRINTER,RECSIZE=80,IOERR GOTO IOERR3
000970
000980 /* SEE IF THE SPECIFIED FILE EXISTS */
000990
001000 CALL "LIMITS"(FILE$,LIB$,IVOL$,CODE$,MSG$)
001010 IF CODE$="Y" THEN ZZZ
001020 IF CODE$="N" THEN GOSUB'7(F$,HEX(91),HEX(81),HEX(81))
001030 IF CODE$="V" THEN GOSUB'7(STR(MSG$,3),HEX(81),HEX(81),HEX(91))
001040 IF CODE$="L" THEN GOSUB'7(STR(MSG$,3),HEX(81),HEX(91),HEX(81))
001050 IF CODE$="E" THEN GOSUB'7(STR(MSG$,3),HEX(81),HEX(81),HEX(81))
001060 GOTO FILMNU
001080
001090 /* IS THIS REALLY A PHRASE SUBSTITUTION LIST? */
001100
001110 ZZZ:
001120 I%=0
001130 CALL "READFOR" ADDR(FILE$,LIB$,IVOL$,I%,"RS",RS%,R%)

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 16:17 11/13/84 Page 3  
Input File is CRES@PSP in Library CRES@SRC on Volume CRES02

```

001140 IF RS%<>70 THEN IOERR2 RT
001150 RT
001160 /* IF WE FALL HERE, IT MUST BE TIME TO OPEN FILE */ RT
001170 RT
001180 OPEN NOGETPARM #1,SHARED, FILE= FILE$, LIBRARY = LIB$, : RT
001190 VOLUME = IVOL$ RT
001200 RT
001210 /* This section of code generates the second screen, which asks */ RT
001220 /* for the particular word interval and its corresponding */ RT
001230 /* print operation. The user is given the option of returning */ RT
001240 /* to the previous screen to select a new word file to print, */ RT
001250 /* or of returning to the substitution list menu. */ JM
001260 RT
001270 WORDEQ$="phrase = " RT
001280 RT
001290 ER$ = " " RT
001300 CWMENU: RT
001310 WORD$ = " " RT
001320 LPP% =50 RT
001330 PF15$ ="PF15" RT
001340 PF16$ ="PF16" RT
001350 SBTITLE$ =" " RT
001360 SBTITL1$ =" " RT
001370 BALONEY: RT
001380 STARTPH$ ="FIRST" RT
001390 ENDPH$ ="LAST" RT
001400 TP$ ="TERMINAL" RT
001410 PRINTSCR: RT
001420 ACCEPT : RT
001430 AT (01,03), : RT
001440 "CRES: Phrase Substitution List Print Subsystem", : RT
001450 AT (01,66), : RT
001460 "(Print Screen)", : RT
001470 AT (02,03), : RT
001480 "===== : RT
001490 =====", : RT
001500 AT (04,13), : RT
001510 "To print a range of lines, fill in the information below.", : RT
001520 AT (06,02), FAC(HEX(85)), : RT
001530 ER$, : RT
001540 AT (08,22), : RT
001550 "Subtitle of list =", : RT
001560 AT (08,42), FAC(HEX(81)), SBTITLE$, CH(20), : RT
001570 AT (09,42), FAC(HEX(81)), SBTITL1$, CH(20), : RT
001580 AT (12,22), : RT
001590 "Starting phrase =", : RT
001600 AT (12,42), FAC(HEX(81)), STARTPH$, CH(24), : RT
001610 RANGE ( ":", " ", "cccccccccccc"), : RT
001620 AT (13,22), : RT
001630 "Ending phrase =", : RT
001640 AT (13,42), FAC(HEX(81)), ENDPH$, CH(24), : RT
001650 RANGE ( ":", " ", "cccccccccccc"), : RT
001660 AT (14,22), : RT
001670 "Lines per page =", : RT
001680 AT (14,42), FAC(HEX(82)), LPP%, PIC(##), : RT
001690 AT (17,22), : RT
001700 "Print at (TERMINAL/PRINTER) =", : RT

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 16:17 11/13/84 Page 4  
Input File is CRES@PSP in Library CRES@SRC on Volume CRES02

```

001710      AT (17,53), FAC(HEX(81)), TP$, CH(08),      : RT
001720      AT (20,32),      : RT
001730 "Press",      : RT
001740      AT (20,38), FAC(HEX(85)), EN$, CH(05),      : RT
001750      AT (20,44),      : RT
001760 "to print",      : RT
001770      AT (21,16),      : RT
001780 "Press",      : RT
001790      AT (21,22), FAC(HEX(84)), PF15$, CH(04),      : RT
001800      AT (21,27),      : RT
001810 "to select another Phrase Substitution List",      : RT
001820      AT (22,17),      : RT
001830 "Press",      : RT
001840      AT (22,23), FAC(HEX(84)), PF16$, CH(04),      : RT
001850      AT (22,28),      : RT
001860 "to return to the Substitution List Menu",      : RT
001870 KEYS(BIN(0)&BIN(15)&BIN(16)&BIN(31)&BIN(32)),      : RT
001880 ON (BIN(15)&BIN(31)&BIN(16)&BIN(32)) GOTO NEW,NEW,EXIT,EXIT      : RT
001890      IF STR(TP$,1,1) = "P" THEN NEXT11      RT
001900      IF STR(TP$,1,1) <> "T" THEN TERMPR      RT
001910 NEXT11:      IF LPP% < 51 THEN NEXT13      RT
001920      ER$="Lines per page' can be at most 50."      RT
001930      GOTO PRINTSCR      RT
001940 IOERR1: PRINT      RT
001950      PRINT BELL,HEX(94),ERR1$      RT
001960      FOR I=1 TO 200000 :NEXT I      RT
001970      GOTO FILMNU      RT
001980      RT
001990 IOERR2: MSG$=ERR1$      RT
002000      M$= HEX(84)      RT
002010      GOTO FILMNU      RT
002020 IOERR3: PRINT      RT
002030      PRINT BELL, HEX(94), ERR2$      RT
002040      FOR I=1 TO 500000 :NEXT I      RT
002050      CLOSE #2      RT
002060      GOTO BALONEY      RT
002070 TERMPR:      RT
002080 ER$="Print at ' has only legal values TERMINAL or PRINTER."      RT
002090      GOTO PRINTSCR      RT
002100      RT
002110 REX: ER$="      'Starting phrase' comes after 'Ending phrase'."      RT
002120      GOTO PRINTSCR      RT
002150      RT
002160 NEXT13: IF STARTPH$="FIRST" OR ENDPH$="LAST" THEN FORD      JEM
002170      IF STARTPH$>ENDPH$ THEN REX      JEM
002180      RT
002190 /* Now it's time to generate our print file name */      RT
002200      RT
002210 FORD:      DAT$= DATE      JEM
002220      D$= STR(DAT$,3,2)&"/"&STR(DAT$,5,2)&"/"&STR(DAT$,1,2)      RT
002230      T$= STR(TIME,1,2)&":"&STR(TIME,3,2)      RT
002240      RT
002250 DISPLAY      : RT
002260      AT (01,03),      : RT
002270 "CRES: Phrase Substitution List Print Subsystem",      : RT
002280      AT (01,61),      : RT
002290 "(Generation Screen)",      : RT

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 16:17 11/13/84 Page 5  
Input File is CRES@PSP in Library CRES@SRC on Volume CRES02

```

002300      AT (02,03),                                ! RT
002310      "=====                                ! RT
002320      =====",                                ! RT
002330      AT (12,23),                                ! RT
002340      "Output file now being generated!"        RT
002350      PFILE$="PHRSUB##"                        RT
002360      I%=0                                        RT
002370      NEXT14: CONVERT I% TO STR(PFILE$,7,2),PIC(00) RT
002380      I%=I%+1                                      RT
002390      CALL "LIMITS" (PFILE$,SLIB$,SVOL$,CODE$,MSG$) JM
002400      IF CODE$="Y" THEN NEXT14                    RT
002402      IF CODE$="E" THEN ER$ = STR(MSG$,3)        JM
002404      IF CODE$="E" THEN PRINTSCR                JM
002410      OPEN NOGETPARM #2,OUTPUT,FILE=PFILE$,LIBRARY=SLIB$, ! JM
002420      VOLUME=SVOL$,SPACE=8000                    JM
002430      GOSUB TTL PGE                                RT
002440      IF STARTPH$ = "FIRST" THEN STARTPH$ = ALL(HEX(00)) RT
002450      IF ENDPH$ = "LAST" THEN ENDPH$ = ALL(HEX(FF)) RT
002460      PAGE% = 1                                      RT
002470      RECS% = 1                                      RT
002480      READ #1, KEY>=STARTPH$,IOBUFF$            JEM
002490                                                    JEM
002500      /* RECORD MAY IDENTIFY THE FILE, SO SEE IF WE SHOULD READ AGAIN */ JEM
002510                                                    JEM
002520      IF STR(IOBUFF$,1,24)<>" PHRASE SUBSTITUTION" THEN HEAD JEM
002530      READ #1,IOBUFF$,EOD GOTO HOME                JEM
002540      HEAD: GOSUB HEADER                            JEM
002550      IF STR(IOBUFF$,1,24)>ENDPH$ THEN HOME          RT
002560      GOTO KNOT                                      RT
002570      AGAN: READ #1, IOBUFF$,EOD GOTO HOME          RT
002580      IF STR(IOBUFF$,1,24)>ENDPH$ THEN HOME          RT
002590      IF RECS% = 1 THEN GOSUB HEADER                RT
002600      KNOT:                                          RT
002610      WRITE #2, USING CAT6,C$,STR(IOBUFF$,1,24), ! RT
002620      STR(IOBUFF$,25,20),STR(IOBUFF$,45,20)         RT
002630                                                    RT
002640      RECS%=RECS%+1                                  RT
002650      IF RECS%>LPP% THEN RECS%=1                    RT
002660      IF RECS%=1 THEN PAGE%=PAGE%+1                RT
002670      GOTO AGAN                                      RT
002680      HOME: WRITE #2, USING DOG, C$                 RT
002690      WRITE #2, USING DOG, C$                        RT
002700      WRITE #2, USING CAT8, C$                       RT
002710      CLOSE#2                                        RT
002720      IF STR(IPS$,1,1) ="P" THEN MESS              RT
002730      CLOSE WS                                        RT
002740      MSG2$ = "CANCEL Processing"                   RT
002750      MSK$=HEX(00)                                  RT
002760      CMSG$ = " "                                     RT
002770      CALL "LINK" ADDR("DISPFILE","P",RLIB$,RVOL$,3%,PFILE$, ! JM
002780      SLIB$,SVOL$, "C",MSG2$,27%, "H",MSK$,CMSG$,255%,C%,R%) JM
002790      ER$ = " "                                       RT
002800      IF C%<>0 OR R%<>0                                ! RT
002810      THEN ER$ = STR(CMSG$,141)                    RT
002820      GOTO CWMENU                                    RT
002830                                                    RT
002840      MESS: ER$="      File "&PFILE$& " in Library "& SLIB$ & " on "& ! JM

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 16:17 11/13/84 Page 6  
Input File is CRES@PSP in Library CRES@SRC on Volume CRES02

```

002850 SVOL$ &" generated." JM
002860 GOTO BALONEY RT
002870 RT
002880 NEW: PRINT PAGE, AT(12,28),HEX(84);"Closing last file" RT
002890 FOR I= 1 TO 50000 :NEXT I RT
002900 FILE$=" "
002910 CLOSE #1 RT
002920 GOTO AGAIN RT
002930 RT
002940 TTL_PGE: RT
002950 C$=HEX(8001) RT
002960 WRITE #2, USING DOG, C$ RT
002970 C$ = HEX(0015) RT
002980 WRITE #2, USING DOG, C$ RT
002990 C$ = HEX(0001) RT
003000 WRITE #2, USING DOG1, C$ RT
003010 WRITE #2, USING DOG2, C$ RT
003020 WRITE #2, USING DOG0, C$,D$,T$ RT
003030 WRITE #2, USING DOG3, C$ RT
003040 WRITE #2, USING DOG2, C$ RT
003050 WRITE #2, USING DOG1, C$ RT
003060 WRITE #2, USING DOG2, C$ RT
003070 IF SBITLE$ = " * THEN GERRY RT
003080 WRITE #2, USING DOG4, C$, SBITLE$ RT
003090 IF SBITLE$ <> " * THEN WRITE #2, USING : RT
003100 DOG10, C$, SBITLE$ RT
003110 WRITE #2, USING DOG2, C$ RT
003120 GERRY: WRITE #2, USING DOG5, C$, FILE$ RT
003130 WRITE #2, USING DOG2, C$ RT
003140 WRITE #2, USING DOG6, C$, LIB$ RT
003150 WRITE #2, USING DOG2, C$ RT
003160 WRITE #2, USING DOG7, C$, IVOL$ RT
003170 WRITE #2, USING DOG2, C$ RT
003180 WRITE #2, USING DOG8, C$, STARTPH$ RT
003190 WRITE #2, USING DOG2, C$ RT
003200 WRITE #2, USING DOG9, C$, ENDPH$ RT
003210 WRITE #2, USING DOG2, C$ RT
003220 WRITE #2, USING DOG1, C$ RT
003230 RETURN RT
003240 RT
003250 HEADER: RT
003260 IF PAGE% = 1 THEN FIRST RT
003270 C$ = HEX(0001) RT
003280 WRITE #2, USING DOG, C$ RT
003290 FIRST: C$ = HEX(8001) RT
003300 WRITE #2, USING DOG, C$ RT
003310 C$ = HEX(0001) RT
003320 IF SBITLE$ = " : RT
003330 THEN WRITE #2, USING CAT,C$,D$,T$,PAGE% : RT
003340 ELSE GOTO FOUT RT
003350 GOTO ALRITE RT
003360 FOUT: WRITE #2, USING CAT1,C$,SBITLE$,D$,T$,PAGE% RT
003370 IF SBITLE$ <> " : RT
003380 THEN WRITE #2, USING CAT7,C$,SBITLE$ RT
003390 ALRITE: WRITE #2, USING CAT2, C$, FILE$, LIB$, IVOL$ RT
003400 WRITE #2, USING CAT3, C$ RT
003410 WRITE #2, USING CAT4, C$ RT

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 16:17 11/13/84 Page 7  
Input File is CRES@PSP in Library CRES@SRC on Volume CRES02

```

003420                                     RT
003430      WRITE #2, USING CAT3, C$      RT
003440      RETURN                        RT
003450                                     RT
003460 DOG0:###      !      !      RT
003470      #####      !      RT
003480 DOG: ###      RT
003490 DOG1:###      !-----!      RT
003500 -----!      RT
003510 DOG2:###      !      RT
003520      !      RT
003530 DOG3:###      !      ***** PHRASE SUBSTITUTION LIST      RT
003540 *****      !      RT
003550 DOG4:###      !      TITLE OF LIST: #####!      RT
003560 #      !      RT
003570 DOG5:###      !      INPUT FILE: #####      RT
003580      !      RT
003590 DOG6:###      !      INPUT LIBRARY: #####      RT
003600      !      RT
003610 DOG7:###      !      INPUT VOLUME: #####      RT
003620      !      RT
003630 DOG8:###      !      STARTING PHRASE: #####!      RT
003640 #####      !      RT
003650 DOG9:###      !      ENDING PHRASE: #####!      RT
003660 #####      !      RT
003670 DOG10:###      !      #####!      RT
003680 ##      !      RT
003690 CAT:### #####      !      RT
003700      PAGE #####      RT
003710 CAT1:### TITLE OF LIST : #####      RT
003720 #####      PAGE #####      RT
003730 CAT7:### #####      RT
003740 CAT2:### Input File is ##### in Library ##### on Volume ###!      RT
003750 ###      RT
003760 CAT3:###=====!      RT
003770 =====      RT
003780 CAT4:### PHRASE      First Substitute      Sec!      RT
003790 ond Substitute      RT
003800      RT
003810 CAT6:### #####      RT
003820 #####      RT
003830 CAT8:### !! END OF LISTING !!      RT
003840      RT
003850 /* THIS FUNCTION PRINTS OUT ERROR MESSAGES AND */      RT
003860 /* SETS UP FLASHING FAC'S IN APPROPRIATE PLACES */      RT
003870 DEF FN'7(MSG$,B$,C$,D$)      RT
003880      PRINT      RT
003890      M$=HEX(94)      RT
003900      G$=B$      RT
003910      L$=C$      RT
003920      V$=D$      RT
003930 RETURN      RT
003940      RT
003950 EXIT: CLOSE #1      RT
003960 OUT: /* RETURNING TO SUBSTITUTION LIST MENU */      JM

```

Technical Report 86-001

THIS PAGE INTENTIONALLY LEFT BLANK.

Print Merged Word and Phrase Substitute List (CRES@SBP)

CRES@SBP in library CRES@SRC is the BASIC language program to print the merged Word Substitute List and Phrase Substitute List. The object language version is CRES@SBP in library CRES@SAV.

This program first allows the user to specify the Word and Phrase Substitute Lists to be printed. Then the program enables the user to print out either the entire merged Substitute List or a specified portion. This printout can be displayed at the terminal or sent to the printer. An optional subtitle of up to two lines can be included for the printout, and the number of lines per page may be specified as some value less than the default value of 50.

A complete program listing follows. Appendix C contains a sample flowchart.

Technical Report 86-001

THIS PAGE INTENTIONALLY LEFT BLANK.

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 16:20 11/13/84 Page 1  
Input File is CRES\$SBP in Library CRES\$SRC on Volume CRES02

```

000010 /* This is the CRES WORD & PHRASE SUBSTITUTION PRINT PROGRAM */ RT
000020 /* */ RT
000030 /*          AUTHORS: Rose Tank and John Mears */ RT
000040 /*          Date:    October 12, 1983 */ RT
000050 /* */ RT
000060 /* */ RT
000070 /* The following code sets up a screen, which asks for the */ RT
000080 /* names of the files containing the word and phrase sub- */ JM
000081 /* stitution lists to be printed. The screen also provides */ JM
000090 /* the opportunity to change the library and volume names */ JM
000100 /* which are set to the appropriate defaults through the */ JM
000110 /* "Extract" system routine. */ JM
000120 RT
000130 RT
000140 DIM WORDS$24,ERR1$53,CODE$1,MSG$54,F$40,CMS$255,MSG2$40 JM
000150 DIM IOBUFFP$70,IOBUFFW$80,ERR2$54,OPEN$(2),FIL$34 JM
000160 DIM HEADER$40,L1$1,V1$1,G1$1,L2$1,V2$1,G2$1,B$1,C$2,D$8,M$1,MSK$2 RT
000170 DIM SBTITLE$20,SBTITL1$20,SLIB$8,SVOL$6,RLIB$8,RVOL$6 JM
000180 DIM ER$ 79 RT
000190 DIM STARTPH$ 24 RT
000200 DIM ENDPH$ 24 RT
000210 DIM TP$ 08 RT
000220 DIM PF15$ 04 RT
000230 DIM PF16$ 04 RT
000240 RT
000250 F$= "Filename not found on (library/volume)." RT
000260 ERR1$= "I/O ERROR - Please specify a WORD SUBSTITUTION list." RT
000270 ERR2$= "I/O ERROR - Please specify a PHRASE SUBSTITUTION list." RT
000280 ERR3$= "I/O ERROR - Internal format not matched." RT
000290 AGAIN: G1$, G2$=HEX(81) RT
000300 MSG$= " :L1$, L2$=HEX(81) RT
000310 V1$,V2$=HEX(81): M$=HEX(8D) RT
000320 RT
000330 DIM TYPE1$ 22 RT
000340 DIM TYPE2$ 24 RT
000350 DIM WFILE$ 08 RT
000360 DIM PFILE$ 08 RT
000370 DIM WLIB$ 08 RT
000380 DIM PLIB$ 08 RT
000390 DIM WVOL$ 06 RT
000400 DIM PVOL$ 06 RT
000410 DIM ENTER$ 05 RT
000420 HEADER$ = "CRES SUBSTITUTION LISTS PRINT SUBSYSTEM" RT
000430 FIL$ = "word and Phrase Substitution Lists" RT
000440 TYPE1$ = "WORD SUBSTITUTION LIST" RT
000450 TYPE2$ = "PHRASE SUBSTITUTION LIST" RT
000460 PFILE$ = "PHRSBLST" RT
000461 WFILE$ = "WRDSBLST"
000470 CALL "EXTRACT" ADDR("IL",WLIB$,"IL",PLIB$,"IV",WVOL$,"IV",PVOL$, : JM
000480 "SL",SLIB$,"SV",SVOL$,"RL",RLIB$,"RV",RVOL$) JM
000490 ENTER$ = "ENTER" RT
000500 PF16$ = "PF16" RT
000510 FILMNU: RT
000520 ACCEPT : RT
000530 AT (01,03), : RT
000540 "CRES: Merge Substitution Lists Print Subsystem (File Sele: RT
000550 ction Screen)", : RT

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 16:20 11/13/84 Page 2  
Input File is CRESE\$BP in Library CRESE\$SRC on Volume CRE\$02

```

000560         AT (02,03),                                : RT
000570 "=====": RT
000580 "=====": RT
000590         AT (04,11),                                : RT
000600 "Welcome to the",                                : RT
000610         AT (04,27), FAC(HEX(85)),HEADER$           , CH(39), : RT
000620         AT (07,12),                                : RT
000630 "Please specify the",                                : RT
000640         AT (07,31), FAC(HEX(84)),FIL$             , CH(34), : RT
000650         AT (08,26),                                : RT
000660 "you want to jointly print. ",                    : RT
000670         AT (12,07), FAC(HEX(85)), TYPE1$           , CH(22), : RT
000680         AT (12,52), FAC(HEX(85)), TYPE2$           , CH(24), : RT
000690         AT (14,09),                                : RT
000700 "File      =",                                : RT
000710         AT (14,19), FAC(G1$),           WFILES$     , CH(08), : RT
000720         RANGE ( ":", " ", "#####"),            : RT
000730         AT (14,55),                                : RT
000740 "File      =",                                : RT
000750         AT (14,66), FAC(G2$),           PFILES$     , CH(08), : RT
000760         RANGE ( ":", " ", "#####"),            : RT
000770         AT (15,09),                                : RT
000780 "Library =",                                : RT
000790         AT (15,19), FAC(L1$),           WLIB$       , CH(08), : RT
000800         AT (15,55),                                : RT
000810 "Library =",                                : RT
000820         AT (15,66), FAC(L2$),           PLIB$       , CH(08), : RT
000830         AT (16,09),                                : RT
000840 "Volume  =",                                : RT
000850         AT (16,19), FAC(V1$),           WVOL$       , CH(06), : RT
000860         AT (16,55),                                : RT
000870 "Volume  =",                                : RT
000880         AT (16,66), FAC(V2$),           PVOL$       , CH(06), : RT
000890         AT (18,20), FAC(M$), MSG$,                : RT
000900         AT (21,30),                                : RT
000910 "Press",                                : RT
000920         AT (21,36), FAC(HEX(85)),           ENTER$    , CH(05), : RT
000930         AT (21,42),                                : RT
000940 "to continue",                                : RT
000950         AT (22,18),                                : RT
000960 "Press",                                : RT
000970         AT (22,24), FAC(HEX(84)),           PF16$      , CH(04), : RT
000980         AT (22,29),                                : RT
000990 "to return to the Substitution Menu",            : RT
001000 KEYS(BIN(0)&BIN(16)&BIN(32)),                    :
001010 ON (BIN(16)&BIN(32)) GOTO OUT                    : RT
001020                                                    : RT
001030         G1$, G2$=HEX(81)                            JM      :
001040         L1$, L2$=HEX(81)                            JM      :
001050         V1$, V2$=HEX(81)                            JM      :
001060                                                    : RT
001070 /* SET UP THE USER FILE BLOCK */                : RT
001080                                                    : RT
001090 SELECT #1,"WRDSUB",VARC,INDEXED,RECSIZE=80,KEYPOS=1,KEYLEN=20, : RT
001100         IOERR GOTO IOERR1                            : RT
001110                                                    : RT
001120 SELECT #2,"PHRASE",VARC,INDEXED,RECSIZE=70,KEYPOS=1,KEYLEN=24, : RT

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 16:20 11/13/84 Page 3  
Input File is CRES@SBP in Library CRES@SRC on Volume CRES02

```

001130          IOERR GOTO IOERR1                      RT
001140                                              RT
001150 SELECT #3,"PRTFILE",PRINTER,RECSIZE=80,IOERR GOTO IOERR3 RT
001160                                              RT
001170 /* SEE IF THE SPECIFIED FILES EXISTS */          JM
001180                                              RT
001190 CALL "LIMITS"(WFILE$,WLIB$,WVOL$,CODE$,MSG$)      RT
001200 IF CODE$="Y" THEN CKFILE2                          RT
001210 IF CODE$="N" THEN GOSUB'7(F$,HEX(91),HEX(81),HEX(81)) RT
001220 IF CODE$="V" THEN GOSUB'7(STR(MSG$,3),HEX(81),HEX(81),HEX(91)) RT
001230 IF CODE$="L" THEN GOSUB'7(STR(MSG$,3),HEX(81),HEX(91),HEX(81)) RT
001240 IF CODE$="E" THEN GOSUB'7(STR(MSG$,3),HEX(81),HEX(81),HEX(81)) RT
001250 GOTO FILMNU                                      RT
001260 CKFILE2:                                          RT
001270 CALL "LIMITS"(PFILE$,PLIB$,PVOL$,CODE$,MSG$)    RT
001280 IF CODE$="Y" THEN ZZZ                            RT
001290 IF CODE$="N" THEN GOSUB'8(F$,HEX(91),HEX(81),HEX(81)) RT
001300 IF CODE$="V" THEN GOSUB'8(STR(MSG$,3),HEX(81),HEX(81),HEX(91)) RT
001310 IF CODE$="L" THEN GOSUB'8(STR(MSG$,3),HEX(81),HEX(91),HEX(81)) RT
001320 IF CODE$="E" THEN GOSUB'8(STR(MSG$,3),HEX(81),HEX(81),HEX(81)) RT
001330 GOTO FILMNU                                      RT
001340                                              RT
001350 /* ARE THESE REALLY WORD & PHRASE SUBSTITUTION LISTS? */ JM
001360                                              RT
001370 ZZZ:                                              RT
001380 I% = 0                                           RT
001390 CALL "READPDR" ADDR(WFILE$,WLIB$,WVOL$,I%,"RS",RS%,R%) RT
001400 IF RS% < > 80 THEN IOERR1                        RT
001410                                              RT
001420 CALL "READPDR" ADDR(PFILE$,PLIB$,PVOL$,I%,"RS",RS%,R%) RT
001430 IF RS% < > 70 THEN IOERR2                        RT
001440                                              RT
001450 /* IF WE FALL HERE, IT MUST BE TIME TO OPEN FILES */ JM
001460                                              RT
001470          IF OPEN%(1)=1 THEN CLOSE #1              RT
001480          OPEN NOGETPARM #1,SHARED, FILE= WFILE$, LIBRARY = WLIB$, : RT
001490          VOLUME = WVOL$                          RT
001500          OPEN%(1)=1                               RT
001510                                              RT
001520          IF OPEN%(2)=1 THEN CLOSE #2              RT
001530          OPEN NOGETPARM #2,SHARED, FILE= PFILE$, LIBRARY = PLIB$, : RT
001540          VOLUME = PVOL$                          RT
001550          OPEN%(2)=1                               RT
001560                                              RT
001570 /* This section of code generates the second screen, which asks */ RT
001580 /* for the particular word interval and its corresponding */ RT
001590 /* print operation. The user is given the option of returning */ RT
001600 /* to the previous screen to select new substitution files to */ JM
001610 /* print, or of returning to the substitution list menu. */ JM
001620                                              RT
001630                                              RT
001640 ERS$ = " "                                         RT
001650 CWMENU:                                          RT
001660          WORD$ = " "                               RT
001670          LPP%          =50                         RT
001680          PF15$         ="PF15"                     RT
001690          SBFILE$       =" "                         RT

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 16:20 11/13/84 Page 4  
Input File is CRES@SBP in Library CRES@SRC on Volume CRES02

```

001700      SBFTITLE$      = " "      RT
001710 BALONEY:      RT
001720      WF,PF,w=0      RT
001730      STARTPH$      ="FIRST"      RT
001740      ENDPH$      ="LAST"      RT
001750      TP$      ="TERMINAL"      RT
001760 PRINTSCR:      RT
001770 ACCEPT      : RT
001780      AT (01,03),      : RT
001790 "CRES: Merge Substitution Lists Print Subsystem",      : RT
001800      AT (01,66),      : RT
001810 "(Print Screen)",      : RT
001820      AT (02,03),      : RT
001830 "=====":      RT
001840 "=====":      RT
001850      AT (04,13),      : RT
001860 "To print a range of lines, fill in the information below.",      : RT
001870      AT (06,02), FAC(HEX(85)),      : RT
001880 ER$,      : RT
001890      AT (08,20),      : JM
001900 "Subtitle of list      =",      : RT
001910      AT (08,46), FAC(HEX(81)), SBFTITLE$, CH(20),      : JM
001920      AT (09,46), FAC(HEX(81)), SBFTITLE$, CH(20),      : JM
001930      AT (12,20),      : JM
001940 "Starting word or phrase =",      : RT
001950      AT (12,46), FAC(HEX(81)), STARTPH$, CH(24),      : JM
001960      RANGE ( ":", " ", "#####"),      : RT
001970      AT (13,20),      : JM
001980 "Ending word or phrase =",      : RT
001990      AT (13,46), FAC(HEX(81)), ENDPH$, CH(24),      : JM
002000      RANGE ( ":", " ", "#####"),      : RT
002010      AT (14,20),      : JM
002020 "Lines per page      =",      : RT
002030      AT (14,46), FAC(HEX(82)), LPP$, PIC(##),      : JM
002040      AT (16,20),      : JM
002050 "Print at (TERMINAL/PRINTER) =",      : RT
002060      AT (16,51), FAC(HEX(81)), TP$, CH(08),      : JM
002070      AT (21,32),      : JM
002080 "Press",      : RT
002090      AT (21,38), FAC(HEX(85)), ENTER$, CH(05),      : JM
002100      AT (21,44),      : JM
002110 "to print",      : RT
002120      AT (22,17),      : JM
002130 "Press",      : RT
002140      AT (22,23), FAC(HEX(84)), PF15$, CH(04),      : JM
002150      AT (22,28),      : JM
002160 "to select two more Substitution Lists",      : RT
002170      AT (23,16),      : JM
002180 "Press",      : RT
002190      AT (23,22), FAC(HEX(84)), PF16$, CH(04),      : JM
002200      AT (23,27),      : JM
002210 "to return to the Substitution List Menu",      : RT
002220 KEYS(BIN(0)&BIN(15)&BIN(16)&BIN(31)&BIN(32)),      : RT
002230 ON (BIN(15)&BIN(31)&BIN(16)&BIN(32)) GOTO NEW,NEW,EXIT,EXIT      : RT
002240 ER$=" "      RT
002250      IF STR(TP$,1,1) = "P" THEN NEXT11      RT
002260      IF STR(TP$,1,1) <> "T" THEN TERMPR      RT

```

# Technical Report 06-001

Wang VS Integrated Editor - Version 6.10.00 16:20 11/13/84 Page 5  
Input File is CRES0S0P in Library CRES0SRC on Volume CRES02

```

002270 NEXT11:      IF LPP# < 51 THEN NEXT13      RT
002280              ER$="Lines per page' can be at most 50."  RT
002290              GOTO PRINTSCR                  RT
002300              RT
002310 IOERR1:      MSG$=ERR1$                    RT
002320              M$= HEX(84)                    RT
002330              GOTO FILMNU                     RT
002370              RT
002380 IOERR2:      MSG$=ERR2$                    RT
002390              M$= HEX(84)                    RT
002400              GOTO FILMNU                     RT
002410              RT
002420 IOERR3:      PRINT                          RT
002430              PRINT BELL, HEX(94), ERR3$      RT
002440              FOR I=1 TO 500000 :NEXT I      RT
002450              CLOSE #3                        RT
002460              GOTO BALONEY                   RT
002470 TERMPR:      RT
002480 ER$="Print at ' has only legal values TERMINAL or PRINTER." RT
002490              GOTO PRINTSCR                  RT
002500              RT
002510 REX: ER$="      'Starting word' comes after 'Ending word'." RT
002520              GOTO PRINTSCR                  RT
002540              RT
002550 NEXT13: IF STARTPH$="FIRST" OR ENDPH$="LAST" THEN FORD JM
002560              IF STARTPH$>ENDPH$ THEN REX      JM
002570              RT
002580 /* Now it's time to generate our print file name */ RT
002590              RT
002600 FORD:      DAT$= DATE                        JM
002610              D$= STR(DAT$,3,2)&"/"&STR(DAT$,5,2)&"/"&STR(DAT$,1,2) RT
002620              T$= STR(TIME,1,2)&":"&STR(TIME,3,2) RT
002630              RT
002640 DISPLAY      : RT
002650              AT (01,03),                      : RT
002660 "CRES: Merge Substitution List Print Subsystem", : RT
002670              AT (01,61),                      : RT
002680 "(Generation Screen)",                        : RT
002690              AT (02,03),                      : RT
002700 "=====": RT
002710 "=====": RT
002720              AT (12,23),                      : RT
002730 "Output file now being generated!"           RT
002740 PFIL$="MERSUB##"                             RT
002750              I# = 0                           JM
002760 NEXT14: CONVERT I# TO STR(PFIL$,7,2),PIC(00)  RT
002770              I# = I# + 1                      RT
002780              CALL "LIMITS" (PFIL$,SLIB$,SVOL$,CODE$,MSG$) JM
002790              IF CODE$="Y" THEN NEXT14          RT
002800              IF CODE$="E" THEN ER$ = STR(MSG$,3) RT
002810              IF CODE$="E" THEN PRINTSCR        RT
002820              OPEN NOGETPARM #3,OUTPUT,FILE=PFIL$,LIBRARY=SLIB$, : JM
002830              VOLUME=SVOL$,SPACE=8000          JM
002840              GOSUB PFL PGE                     RT
002850              IF STARTPH$ = "FIRST" THEN STARTPH$ = "ZZZ" RT
002860              IF ENDPH$ = "LAST" THEN ENDPH$ = ALL(HEX(FF)) RT
002870              PAGE# = 1                         RT

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 16:20 11/13/84 Page 6  
Input File is CRES@3BP in Library CRES@SRC on Volume CRES02

```

002880      RECS% = 1                                RT
002890      READ #1, KEY)=STARTPH$,IOBUFFW$          RT
002900      READ #2, KEY)=STARTPH$,IOBUFFP$          RT
002910      IF STR(IOBUFFW$,20)<STR(IOBUFFP$,20) THEN W=1 ELSE W=0 RT
002920      GOSUB HEADER                               RT
002930      IF (STR(IOBUFFW$,20)>STR(ENDPH$,20)) AND    ! RT
002940      (STR(IOBUFFP$,24)>STR(ENDPH$,24)) THEN HOME RT
002950                                              RT
002960 KNOT:                                         RT
002970      IF W=1 THEN                                ! RT
002980      WRITE #3, USING WDIMG5,C$,STR(IOBUFFW$,1,20), ! RT
002990      STR(IOBUFFW$,21,20),STR(IOBUFFW$,41,15)    ! RT
003000      ELSE                                         ! RT
003010      WRITE #3, USING WDIMG6,C$,STR(IOBUFFP$,1,24), ! RT
003020      STR(IOBUFFP$,25,20),STR(IOBUFFP$,45,15)    RT
003030      RECS%=RECS%+1                              RT
003040      IF RECS%>LPP% THEN RECS%=1                RT
003050      IF RECS%=1 THEN PAGE%=PAGE%+1              RT
003060      IF WF<>1 AND W=1 THEN READ #1, IOBUFFW$,EOD GOTO FINW RT
003070      IF PF<>1 AND W=0 THEN READ #2, IOBUFFP$,EOD GOTO FINP RT
003090      IF STR(IOBUFFW$,20)>ENDPH$ THEN WF=1        RT
003100      IF STR(IOBUFFP$,24)>ENDPH$ THEN PF=1        RT
003110      IF WF=1 AND PF=1 THEN HOME                 RT
003120      IF STR(IOBUFFW$,20)<STR(IOBUFFP$,20) THEN W=1 ELSE W=0 RT
003130      IF RECS% = 1 THEN GOSUB HEADER             RT
003140      GOTO KNOT                                   RT
003150 HOME: WRITE #3, USING DOG, HEX(0002)           RT
003170      WRITE #3, USING WDIMG8, C$                 RT
003180      CLOSE#3                                       RT
003190      IF STR(TP$,1,1) ="P" THEN MESS            RT
003200      CLOSE WS                                     RT
003210      MSG2$ = "CANCEL Processing"               RT
003220      MSK$=HEX(00)                              RT
003230      CMSC$ = " "                               RT
003240 CALL "LINK" ADDR("DISPFILE","P",RLIB$,RVOL$,3%,PFIL$, SLIB$,    ! JM
003250      SVOL$, "C",MSG2$,27%,"H",MSK$,CMSC$,255%,C%,R%)    JM
003260      ER$ = " "                                     RT
003270      IF C%<>0 OR R%<>0                             ! RT
003280      THEN ER$ = STR(CMSC$,141)                   RT
003290      GOTO CWMENU                                  RT
003300                                              RT
003310 MESS: ER$="          File "&PFIL$ & " in Library "& SLIB$ & " on "& ! JM
003320      SVOL$ & " generated."                      JM
003330      GOTO BALONEY                                  RT
003340                                              RT
003350 NEW: PRINT PAGE, AT(12,28),HEX(84);"Closing last file" RT
003360      WF,PF,W =0                                    RT
003370      FOR I= 1 TO 50000 :NEXT I                  RT
003380      CLOSE #1:CLOSE #2 : OPEN%(1),OPEN%(2)=0      RT
003390      GOTO AGAIN                                   RT
003400                                              RT
003410 TTL_PGE:                                       RT
003420      C$=HEX(8001)                                RT
003430      WRITE #3, USING DOG, C$                   RT
003440      C$ = HEX(0015)                              RT
003450      WRITE #3, USING DOG, C$                   RT
003460      C$ = HEX(0001)                              RT

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 16:20 11/13/84 Page 7  
Input File is CRES@SBP in Library CRES@SRC on Volume CRES02

```

003470      WRITE #3, USING DOG1, C$      RT
003480      WRITE #3, USING DOG2, C$      RT
003490      WRITE #3, USING DOG0, C$,D$,T$  RT
003500      WRITE #3, USING DOG3, C$      RT
003510      WRITE #3, USING DOG2, C$      RT
003520      WRITE #3, USING DOG1, C$      RT
003530      WRITE #3, USING DOG2, C$      RT
003540      IF SBTTITLE$ = "                " THEN GERRY  RT
003550      WRITE #3, USING DOG4, C$, SBTTITLE$  RT
003560      IF SBTTITLE$ <> "                " THEN WRITE #3, USING : RT
003570 DOG10, C$, SBTTITLE$  RT
003580      WRITE #3, USING DOG2, C$      RT
003590 GERRY:  WRITE #3, USING DOG5, C$,WFILE$,PFILE$  RT
003600      WRITE #3, USING DOG2, C$      RT
003610      WRITE #3, USING DOG6, C$,WLIB$,PLIB$  RT
003620      WRITE #3, USING DOG2, C$      RT
003630      WRITE #3, USING DOG7, C$,WVOL$,PVOL$  RT
003640      WRITE #3, USING DOG2, C$      RT
003650      WRITE #3, USING DOG8, C$, STARTPH$  RT
003660      WRITE #3, USING DOG2, C$      RT
003670      WRITE #3, USING DOG9, C$, ENDPH$  RT
003680      WRITE #3, USING DOG2, C$      RT
003690      WRITE #3, USING DOG1, C$      RT
003700      RETURN  RT
003710  RT
003720 HEADER:  RT
003730      IF PAGE#=1 THEN FIRST  RT
003740      C$ = HEX(0001)  RT
003750      WRITE #3, USING DOG, C$  RT
003760 FIRST:  C$ = HEX(8001)  RT
003770      WRITE #3, USING DOG, C$  RT
003780      C$ = HEX(0001)  RT
003790      IF SBTTITLE$ = "                " : RT
003800      THEN WRITE #3, USING WDIMG,C$,D$,T$,PAGE# : RT
003810      ELSE GOTO FOUT  RT
003820      GOTO ALRITE  RT
003830 FOUT:  WRITE #3, USING WDIMG1,C$,SBTTITLE$,D$,T$,PAGE#  RT
003840      IF SBTTITLE$ <> "                " : RT
003850      THEN WRITE #3, USING WDIMG7,C$,SBTTITLE$  RT
003860 ALRITE:  WRITE #3, USING WDIMG2, C$,WFILE$,WLIB$, WVOL$  RT
003870      WRITE #3, USING WDIMG9, C$,PFILE$,PLIB$, PVOL$  RT
003880      WRITE #3, USING WDIMG3, C$  RT
003890      WRITE #3, USING WDIMG4, C$  RT
003910      WRITE #3, USING WDIMG3, C$  RT
003920      RETURN  RT
003930  RT
003940 DOG0:### : ##### : RT
003950 ##### : RT
003960 DOG: ### RT
003970 DOG1:### :-----: RT
003980 -----: RT
003990 DOG2:### : : RT
004000 : RT
004010 DOG3:### : ***** MERGE SUBSTITUTION LISTS : RT
004020 ***** : RT
004030 DOG4:### : TITLE OF LIST: #####: RT
004040 ##### : RT

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 16:20 11/13/84 Page 8  
Input File is CRES@SBP in Library CRES@SRC on Volume CRES02

```

004050 DOG5:###          ! WORD FILE : ##### PHRASE FILE : ! RT
004060 #####          ! RT
004070 DOG6:###          ! LIBRARY: ##### LIBRARY: ! RT
004080 #####          ! RT
004090 DOG7:###          ! VOLUME : ##### VOLUME : ! RT
004100 #####          ! RT
004110 DOG8:###          ! STARTING WORD: #####! RT
004120 #####          ! RT
004130 DOG9:###          ! ENDING WORD : #####! RT
004140 #####          ! RT
004150 DOG10:###          ! #####! RT
004160 #####          ! RT
004170 WDIMG:### #####          ! RT
004180 PAGE #####          RT
004190 WDIMG1:### TITLE OF LIST : #####          ! RT
004200 ##### PAGE #####          RT
004210 WDIMG7:### #####          RT
004220 WDIMG2:### WORD File is ##### in Library ##### on Volume #: RT
004230 #####          RT
004240 WDIMG9:### PHRASE File is ##### in Library ##### on Volume: RT
004250 #####          RT
004260 WDIMG3:###=====! RT
004270 =====          RT
004280 WDIMG4:### word or Phrase First Substitute S! RT
004290 econd Substitute          RT
004300 WDIMG5:### #####          ! RT
004310 #####          RT
004320 WDIMG6:### #####          ! RT
004330 #####          RT
004340 WDIMG8:### !!! END OF LISTING !!!          RT
004350          RT
004360 FINW: IF PF=1 THEN HOME          RT
004370 WF=1: IOBUPFW$="ZZZ"          RT
004380 GOTO 3070          RT
004390          RT
004400 FINP: IF WF=1 THEN HOME          RT
004410 PF=1:IOBUPFP$="ZZZ"          RT
004420 GOTO 3090          RT
004430          RT
004440 /* THIS FUNCTION PRINTS OUT ERROR MESSAGES AND */          RT
004450 /* SETS UP FLASHING FAC'S IN APPROPRIATE PLACES */          RT
004460 DEF FN'7(MSG$,B$,C$,D$)          RT
004470 PRINT          RT
004480 M$=HEX(94):G2$,L2$,V2$=HEX(81)          RT
004490 G1$=B$          RT
004500 L1$=C$          RT
004510 V1$=D$          RT
004520 RETURN          RT
004530          RT
004540 DEF FN'8(MSG$,B$,C$,D$)          RT
004550 PRINT          RT
004560 M$=HEX(94):G1$,L1$,V1$=HEX(81)          RT
004570 G2$=B$          RT
004580 L2$=C$          RT
004590 V2$=D$          RT
004600 RETURN          RT
004610 EXIT: CLOSE #1          RT

```

Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 16:20 11/13/84 Page 9  
Input File is CRESESBP in Library CRES@SRC on Volume CRES02

004620 OUT: /\* RETURNING TO SUBSTITUTION LIST MENU \*/

JM

Technical Report 86-001

THIS PAGE INTENTIONALLY LEFT BLANK.

## GLOSSARY SUBSYSTEM

### Create/Maintain Glossary Menu (CRES@BCG)

CRES@BCG in library CRES@PRC is the Procedure Language menu routine that serves as the driver for the Create/Maintain Glossary Subsystem. It enables the user to select by PF key among options to create/maintain the Abbreviation-Acronym Glossary or the Technical Glossary, to print the Abbreviation-Acronym Glossary or the Technical Glossary, or to return to the CRES Master Menu.

A complete listing follows.

Technical Report 86-001

THIS PAGE INTENTIONALLY LEFT BLANK.

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 14:44 11/15/84 Page 1  
Input File is CRES@BCG in Library CRES@PRC on Volume CRES01

```

*
*      PROCEDURE      CRES ABBREVIATION/ACRONYM GLOSSARY MAINTENANCE MENU      JM 000100
*      AUTHOR        ROSE TANK                                              JM 000200
*      DATE          10/06/83                                              JM 000300
*
*      Abstract: This procedure is the CRES ABBREVIATION/ACRONYM          JM 000400
*                  GLOSSARY MAINTENANCE MENU. It allows the user          JM 000500
*                  to select and run any Abbreviation/Acronym             JM 000600
*                  Glossary file option.                                   JM 000700
*
*
*      DECLARE        &USERINPT AS INTEGER      INITIAL 0                  JM 000800
*      DECLARE        &OPTIONERR AS STRING (79) INITIAL "                  JM 000900
*      DECLARE        &OPTIONERR2 AS STRING (79) INITIAL "                  JM 001000
*      DECLARE        &USERNAME AS STRING (24) INITIAL "                    JM 001100
*      DECLARE        &ERRMSG AS STRING (34) INITIAL "                      JM 001200
*      DECLARE        &RUNLIB AS STRING (8) INITIAL "                        JM 001300
*      DECLARE        &RUNVOL AS STRING (6) INITIAL "                        JM 001400
*
*      EXTRACT &USERNAME=USERNAME, &RUNLIB=OUTLIB, &RUNVOL=RUNVOL          JM 001500
*
*      GETUSERINPT: PROMPT PFKEY = &USERINPT CENTER                        JM 001600
*                  DIM "CRES: Glossary Maintenance Subsystem              (G1 001700
*                  ossary Menu Screen)";                                   JM 001800
*                  DIM "=====";                                         JM 001900
*                  DIM " ";                                               JM 002000
*                  DIM " This subsystem will allow you to maintain two glossar JM 002100
*                  ies - one for ";                                       JM 002200
*                  DIM " abbreviations and acronyms, and another as a general JM 002300
*                  glossary.";                                             JM 002400
*                  DIM " Select the", BRIGHT "PF key", "corresponding to your JM 002500
*                  desired operation.";                                     JM 002600
*                  DIM " ";                                               JM 002700
*                  DIM " ",LINE "PF",DIM " ",                               JM 002800
*                  LINE                                                    JM 002900
*                  . ";DIM " ";                                           JM 003000
*                  DIM " ",Bright "1"," - Maintain Abbreviation-Acron JM 003100
*                  ym Glossary";                                           JM 003200
*                  DIM " ",Bright "2"," - Maintain Glossary";             JM 003300
*                  DIM " ",Bright "3"," - Print Abbreviation-Acronym JM 003400
*                  Glossary";                                              JM 003500
*                  DIM " ",Bright "4"," - Print Glossary";               JM 003600
*                  DIM " ";                                               JM 003700
*                  DIM " ",Bright "16"," - Return to CRES Master Menu" JM 003800
*                  ; DIM " "; " "; " "; " "; " "; " "; " "; " "; " "; JM 003900
*                  BRIGHT &OPTIONERR;                                     JM 004000
*                  BLINK &OPTIONERR2;                                     JM 004100
*
*      ASSIGN        &ERRMSG = " "                                       JM 004200
*      ASSIGN        &OPTIONERR = " "                                     JM 004300
*      ASSIGN        &OPTIONERR2 = " "                                    JM 004400
*
*
*      IF &USERINPT EQ 16 GOTO USEREXIT JM 004500
*      IF &USERINPT EQ 32 GOTO USEREXIT JM 004600
*      IF &USERINPT EQ 1 GOTO RUNBCG JM 004700
*      IF &USERINPT EQ 17 GOTO RUNBCG JM 004800
*      IF &USERINPT EQ 2 GOTO RUNGLO JM 004900

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 14:44 11/15/84 Page 2  
Input File is CRES@BCG in Library CRES@PRC on Volume CRES01

IF	&USERINPT EQ 18	GOTO RUNGLO	005900
IF	&USERINPT EQ 3	GOTO RUNPBC	006000
IF	&USERINPT EQ 19	GOTO RUNPBC	006100
IF	&USERINPT EQ 4	GOTO RUNPGL	006200
IF	&USERINPT EQ 20	GOTO RUNPGL	006300
			006400
	ASSIGN &OPTIONERR = "I'm sorry, but your selected option		006500
	is not currently available."		006600
	ASSIGN &OPTIONERR2 = "Please select another topic."		006700
	GOTO GETUSERINPT		006800
*			006900
RUNBCG:	RUN CRES@BCG IN &RUNLIB ON &RUNVOL		007000
	IF RUNBCG EQ 0 GOTO GETUSERINPT		007100
	GOTO USEREXIT		007200
*			007300
RUNGLO:	RUN CRESGLOS IN &RUNLIB ON &RUNVOL		007400
	IF RUNGLO EQ 0 GOTO GETUSERINPT		007500
	GOTO USEREXIT		007600
*			007700
RUNPBC:	RUN CRES@BCP IN &RUNLIB ON &RUNVOL		007800
	IF RUNPBC EQ 0 GOTO GETUSERINPT		007900
	GOTO USEREXIT		008000
*			008100
RUNPGL:	RUN CRESGLOP IN &RUNLIB ON &RUNVOL		008200
	IF RUNPGL EQ 0 GOTO GETUSERINPT		008300
	GOTO USEREXIT		008400
*			008500
USEREXIT:	RETURN		008600

Create/Maintain Abbreviation/Acronym Glossary (CRES@BCG)

CRES@BCG in library CRES@SRC is the BASIC language program to create/maintain the Abbreviation/Acronym Glossary. The object language version is CRES@BCG in library CRES@SAV.

This program first allows the user to specify the Abbreviation/Acronym Glossary to be created or maintained. Then, the program enables the user to add, delete, or modify any given term in the glossary. There is a protection system implemented, and provision is made for clearing the protect bytes, if for some reason the bytes have been left set incorrectly.

Each entry in the Abbreviation/Acronym Glossary can contain up to 20 characters, a flag indicating whether it is an abbreviation or an acronym, and a definition. The data record description is provided in appendix A.

A complete program listing follows. Appendix C contains a sample flowchart.

Technical Report 86-001

THIS PAGE INTENTIONALLY LEFT BLANK.

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 11:25 11/09/84 Page 1  
Input File is CRES@BCG in Library CRES@SRC on Volume CRES02

```

000010 /* CRES ABBREVIATION/ACRONYM GLOSSARY SUBSYSTEM PROGRAM */ RT
000020 /* */ RT
000030 /* AUTHOR : Rose Tank */ JM
000040 /* Date: March 12, 1984 */ RT
000050 /* */ RT
000060 /* Abstract: This program is used to create and maintain the */ JM
000070 /* CRES Aboreviation - Acronym List. */ JM
000071 JM
000080 DIM WORD$20, CODE$1, ER$65, PROTECT$3, MESING$30 JM
000090 DIM F1$20, IOBUFF$190, ER1$50, F2$10 RT
000100 DIM FILE$8, USER$3, USERNAME$24, WRKST$3 RT
000110 DIM HEADER$50, E$1, PF16$4, X$1 JM
000120 DIM ENTER$5, ILIB$8, IVOL$6, PASSWRD$11 JM
000130 DIM IDFL$1, DEF$(3)55 RT
000140 DIM MODE$15, ATTR$18, ENTR$16, TAB$3, BACKTAB$7 RT
000150 ATTR$="CURRENT ATTRIBUTES": ENTR$="LEGAL ATTRIBUTES" RT
000160 TAB$="TAB": BACKTAB$="BACKTAB" RT
000170 ENTER$="ENTER" RT
000180 PF16$="PF16" RT
000190 PF$="PF": OPER$=" Operation" RT
000200 OS="1": TS="2": TH$="3": FT$="14": ST$="16" RT
000210 WELCOMES$="Welcome to the" RT
000220 HEADER$="CRES ABBREVIATION & ACRONYM MAINTENANCE SUBSYSTEM" RT
000230 RT
000240 /* The following two lines are used as formats in the packing */ RT
000250 /* and the unpacking of record field values */ RT
000260 F1$=HEX(A014A001A037A037A037A0030001) RT
000270 F2$=HEX(00BAA0030001) RT
000280 RT
000290 CALL "EXTRACT" ADDR("ID", USER$, "IL", ILIB$, "IV", IVOL$) RT
000300 RT
000310 /* SET UP THE USER FILE BLOCK */ RT
000320 RT
000330 SELECT #1, "ABACGL", VARC, INDEXED, RECSIZE=190, KEYPOS=1, KEYLEN=20 RT
000350 RT
000360 CALL "LIMITS"("ABBRELIST", ILIB$, IVOL$, CODE$) RT
000370 IF CODE$<>"N" THEN 430 JM
000380 OPEN NOGETPARM #1, OUTPUT, FILE="ABBRELIST", LIBRARY=ILIB$, ! RT
000390 VOLUME=IVOL$ RT
000400 IOBUFF$=" ABBREVIATION/ACRONYM GLOSSARY" RT
000410 CLOSE #1 RT
000420 RT
000430 OPEN NOGETPARM #1, SHARED, FILE="ABBRELIST", LIBRARY =ILIB$, ! RT
000440 VOLUME =IVOL$ RT
000450 RT
000460 /* This section of code generates the first screen, which asks */ JM
000470 /* for the particular abbreviation (or acronym) and its corres- */ JM
000475 /* ponding operation - either insert aboreviation (acronym), */ JM
000480 /* delete aboreviation (acronym), or modify aboreviation (acro- */ JM
000485 /* nym). The user is also given the option of returning to the */ JM
000490 /* Glossary Maintenance Menu. */ JM
000510 RT
000520 WORD$, ER$, ER1$=" " : E$=HEX(8C) RT
000530 X$=HEX(81) JM
000540 PASSWRD$=HEX(000102030E101112131E20) JM
000550 RT
000560 CWMENU: RT

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 11:25 11/09/84 Page 2  
Input File is CRES@BOG in Library CRES@SRC on Volume CRES02

```

000570 ACCEPT                                     : RT
000580      AT (01,03),                             : RT
000590 "CRES: Abbreviation and Acronym Subsystem   (Word & Option Sel: RT
000600 ction Screen)",                             : RT
000610      AT (02,03),                             : RT
000620 "=====": RT
000630 =====,                                     : RT
000640      AT (04,10), FAC(HEX(8C)),                 : RT
000650 WELCOMES$,                                     : RT
000660      AT (04,26), FAC(HEX(85)),                 : RT
000670 HEADERS$,                                     : RT
000680      AT (06,05),                             : RT
000690 "Please select an abbreviation or acronym:",    : RT
000700      AT (08,32), FAC(X$),      WORD$,          , CH(20), : JM
000710      AT (10,18), FAC(E$),      ER$,            , CH(60), : RT
000720      AT (11,20), FAC(E$),      ER1$,           , CH(50), : RT
000730      AT (13,05),                                     : RT
000740 "To continue, select the PF key corresponding to the desired oper: RT
000750 ation.",                                     : RT
000760      AT (16,28), FAC(HEX(A5)), PF$,            , CH(02), : RT
000770      AT (16,35), FAC(HEX(AC)), OPER$,          , CH(31), : RT
000780      AT (18,28), FAC(HEX(84)), OS$,             , CH(01), : RT
000790      AT (18,35),                                     : RT
000800 "INSERT abbreviation or acronym",              : RT
000810      AT (19,28), FAC(HEX(84)), IS$,             , CH(1),  : RT
000820      AT (19,35),                                     : RT
000830 "DELETE abbreviation or acronym",              : RT
000840      AT (20,28), FAC(HEX(84)), TH$,             , CH(1),  : RT
000850      AT (20,35),                                     : RT
000860 "MODIFY abbreviation or acronym",              : RT
000870      AT (22,28), FAC(HEX(84)), FT$,             , CH(02), : RT
000880      AT (22,35),                                     : RT
000890 "Protection Override",                         : RT
000900      AT (23,28), FAC(HEX(84)), ST$,             , CH(02), : RT
000910      AT (23,35),                                     : RT
000920 "Return to the Glossary Maintenance Menu",     : RT
000930 KEYS(PASSWRD$),                                : JM
000940      KEY(OPTION)                                : JM
000950 IF OPTION=16 OR OPTION=32 THEN EXIT             : JM
000960                                     : RT
000970 WELCOMES$,HEADERS$=" "                         : RT
000980 ER$,ER1$ =" ": E$=HEX(8C)                     : RT
000990                                     : RT
001000                                     /* Check for confirmation of deletion */ : RT
001010 IF (OPTION=0)AND(DF=1) THEN DEL                : RT
001020                                     /* Check for cancellation of deletion */ : JM
001030 IF DF=1 THEN UPDATE                             : RT
001040                                     /* Check for override of protection bytes */ : JM
001050 IF (OPTION=14) OR (OPTION=30) THEN OVERRIDE     : RT
001060 IF STR(WORDS$,1,1)<>" " THEN REED              : RT
001070                                     : RT
001080 PRINT BELL                                       : RT
001090 ER$="      The first character cannot be blank." : RT
001100 ER1$= "      Please reenter the appropriate word and PF key." : RT
001110 E$= HEX(94)                                     : RT
001120 GOTO CWMENU                                     : RT
001130                                     : RT

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 11:25 11/09/84 Page 3  
Input File is CRES@HCG in Library CRES@SRC on Volume CRES02

```

001140 OVERRIDE: RT
001150 IF NP=1 AND P=1 THEN CHK RT
001160 ER$= "Please check if word exists by selecting an option." JM
001170 E$=HEX(84) RT
001180 GOTO CWMENU RT
001190 RT
001200 /* Protection override is confirmed if the protection was set */ RT
001210 /* by a user not currently logged on or if the current user's */ RT
001220 /* id equals the protection bytes. */ JM
001230 CHK: RT
001240 IF (USER$=PROTECT$) OR (WRKST=-1) THEN REMOVE RT
001250 ER$= "You don't have the clearance to override the protect." RT
001260 ER1$= " " RT
001270 E$= HEX(84) RT
001280 GOTO CWMENU RT
001290 RT
001300 /* Reset the protection byte to blanks */ RT
001310 REMOVE: PROTECT$= " ":P=0 RT
001320 IF WORD$= " " THEN 1080 JM
001330 READ #1, HOLD, KEY=WORD$,IOBUFF$ RT
001340 $PACK (P=F1$) IOBUFF$ FROM WORD$,IDF1$, DEF$(1),DEF$(2), RT
001350 DEF$(3),PROTECT$ RT
001360 REWRITE #1, IOBUFF$, DATA GOTO IOERR2 RT
001370 ER$= " You may now select your option." RT
001380 ER1$= " " RT
001390 E$= HEX(84) RT
001400 GOTO CWMENU RT
001410 RT
001420 IOERR2: PRINT RT
001430 PRINT HEX(94),"Error occurred on file "; FILE$ RT
001440 PRINT HEX(94),"Error code = "; FS(1) RT
001450 STOP "Touch ENTER to return to menu." RT
001460 GOTO CWMENU RT
001470 RT
001480 /* Error message is printed if "ENTER" was hit before the option */ RT
001490 /* was entered */ RT
001500 REED: IF OPTION<>0 THEN SRCH RT
001510 ER$= " Command missing." RT
001520 E$= HEX(94) RT
001530 GOTO CWMENU RT
001540 SRCH: P=1 RT
001550 READ #1,HOLD,KEY=WORD$,IOBUFF$,EOD GOTO NOTFOUND, ! RT
001560 DATA GOTO IOERR2 RT
001570 RT
001580 IF 1 /* The word was found */ RT
001590 RT
001600 /* Unpack and check if currently being modified by another */ RT
001610 RT
001620 $UNPACK (P=F1$) IOBUFF$ TO WORD$,IDF1$,DEF$(1),DEF$(2), ! RT
001630 DEF$(3),PROTECT$ RT
001640 IF PROTECT$= " " THEN NOW RT
001650 CALL "FINDUSER" ADDR(PROTECT$,USERNAME$,WRKST$) RT
001660 CONVERT WRKST$ TO WRKST$, PIC(###) RT
001670 E$= HEX(94) RT
001680 ER$= " Sorry, "& USERNAME$ & " is currently editing " RT
001690 ER1$= " "& WORD$ & " "& " at workstation "& WRKST$ RT
001700 REWRITE #1,IOBUFF$ RT

```

# Technical Report 86-U01

Wang VS Integrated Editor - Version 6.10.00 11:25 11/09/84 Page 4  
Input File is CRES@BCG in Library CRES@SRC on Volume CRES02

```

001710          GOTO CWMENU                      RT
001720                      RT
001730 /* Set the protection bytes */              JM
001740 NOW:      PROTECT$= USER$                    RT
001750          $PACK (F=F1$) IOBUFF$ FROM WORD$, IDFL$, DEF$(1),DEF$(2): RT
001760          ,DEF$(3),PROTECT$                    RT
001770          REWRITE #1, IOBUFF$, DATA GOTO IOERR2 RT
001780                      RT
001790          /* NOW BRANCH TO THE OPTION SELECTED */ RT
001800 OPT:      ON OPTION GOTO INSERT,DELET,MODIFY  RT
001810          ON OPTION-16 GOTO INSERT,DELET,MODIFY RT
001820                      RT
001830 NOTFOUND: NF=0 /* The word was not found in the list */ RT
001840          GOTO OPT                              RT
001850                      RT
001860 INSERT:                      RT
001870          /* NF=0 indicates the word doesn't yet exist so go ahead */ RT
001880          /* and insert. NF=1 indicates the word exists. */ RT
001890          IF NF=1 THEN EXIST                      RT
001900          MODE$= "(Insert Screen)"                RT
001901          MESING$="inserting abbreviation/acronym" JM
001910          IDFL$,DEF$()=" "                      RT
001920          IOBUFF$=" ": PROTECT$= USER$            RT
001930          $PACK (F=F1$) IOBUFF$ FROM WORD$, IDFL$,DEF$(1),DEF$(2), ! RT
001940          DEF$(3),PROTECT$                      RT
001950          WRITE #1, IOBUFF$, DATA GOTO IOERR2  RT
001960          GOTO INS                              RT
001970                      RT
001980 /* Coming from insert with the word currently existing, */ RT
001990 /* print an error message and erase the protection bytes */ JM
002000 EXIST:                      RT
002010          ER$= " " & WORD$ & " already exists in this list." JM
002020          E$= HEX(94)                          RT
002030          READ #1, HOLD, KEY= WORD$,IOBUFF$      RT
002040          PROTECT$=" ":P$=0                      RT
002050          $PACK (F=F1$) IOBUFF$ FROM WORD$, IDFL$,DEF$(1),DEF$(2), ! RT
002060          DEF$(3),PROTECT$                      RT
002070          REWRITE #1, IOBUFF$, DATA GOTO IOERR2 RT
002080          GOTO CWMENU                          RT
002090                      RT
002100 /* DELET,DEL,NODEL- all pertain to the delete option */ RT
002110 /* DELET- gives the user a chance to cancel the deletion */ RT
002120 DELET: IF NF=0 THEN NODEL                      RT
002130          DF=1                                    RT
002140          ER$= " Press ENTER to confirm deletion of "& WORD$ JM
002150          ER1$=" or press PF14 to cancel deletion." JM
002160          X$=HEX(84)                             JA
002170          PASSWRD$=HEX(000E1E)                   JM
002180          E$= HEX(94)                             RT
002190          GOTO CWMENU                          RT
002200                      RT
002210 /* DEL- deletes the entire record */            RT
002220 DEL:      READ #1, HOLD,KEY=WORD$,IOBUFF$      RT
002230          DELETE #1                              RT
002240          DF=0                                    RT
002250          ER$= " "&WORD$&" deleted - Please continue." JM
002260          X$=HEX(81)                             JM

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 11:25 11/09/84 Page 5  
Input File is CRES@BCG in Library CRES@SRC on Volume CRES02

```

002270      PASSWRD$=HEX(000102030E101112131E20)      JM
002280      NF=0      JM
002290      ES= HEX(8C)      JM
002300      WORD$=" "      RT
002310      GOTO CWMENU      RT
002320      RT
002330 /* MODEL- print an error message when the word doesn't exist */      RT
002340 MODEL: ER$= " " & WORD$ & " does not exist in this list."      JM
002350      ES= HEX(94)      RT
002360      GOTO CWMENU      RT
002370      RT
002380 MODIFY: IF NF=1 THEN GETIT      RT
002390      /* if word not in list, */      RT
002400      ER$= "      Modification request denied -"      JM
002410      ER1$= " " & WORD$ & " does not exist in this list."      JM
002420      ES=HEX(94)      /* set up message and fac and */      RT
002430      GOTO CWMENU      /* go back up to previous screen */      RT
002440      RT
002450 GETIT:/*READ #1,HOLD,KEY=WORD$,IOBUFF$,EOD GOTO NOTFOUND, */      RT
002460      /*DATA GOTO IOERR2 */      RT
002470      RT
002480 MODE$      =(Modify Screen)"      RT
002481 MESING$="modifying abbreviation/acronym"      JM
002490 INS:      RT
002500 ER1$      " "      RT
002510      RT
002520 /* This accept statement generates a screen with the attributes */      RT
002530 /* corresponding to each aboveviation (acronym) appropriately */      JM
002535 /* labeled for insert or modification. */      JM
002570      RT
002580 AOK:      RT
002590      RT
002600 ACCEPT      RT
002610      AT (01,03),      RT
002620 "CRES: Abbreviation and Acronym Subsystem",      RT
002630      AT (01,66), FAC(HEX(8C)), MODE$      , CH(15),      RT
002640      AT (02,03),      RT
002650 "=====      RT
002660 =====",      RT
002670      AT (04,06),      RT
002680 "Please enter the appropriate attributes.",      RT
002690      AT (07,33), FAC(HEX(AC)), ATTR$      , CH(18),      RT
002700      AT (07,60), FAC(HEX(AC)), ENTR$      , CH(16),      RT
002710      AT (09,06),      RT
002720 "ABBREVIATION OR ACRONYM:",      RT
002730      AT (09,33), FAC(HEX(85)), WORD$      , CH(20),      RT
002740      AT (11,06),      RT
002750 "IDENTIFICATION FLAG:",      RT
002760      AT (11,33), IDFL$      , CH(01),      RT
002770      AT (11,60),      RT
002780 "B= Abbreviation",      RT
002790      AT (12,60),      RT
002800 "C= Acronym",      RT
002810      AT (14,25), FAC(ES), ER$      , CH(50),      RT
002820      AT (16,06),      RT
002830 "DEFINITION:",      RT
002840      AT (16,20), FAC(HEX(81)), DEF$(1)      , CH(55),      RT

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 11:25 11/09/84 Page 6  
Input File is CRES@BCG in Library CRES@SRC on Volume CRES02

```

002850      AT (17,20), FAC(HEX(81)), DEF$(2)      , CH(55),      : RT
002860      AT (18,20), FAC(HEX(81)), DEF$(3)      , CH(55),      : RT
002870      AT (21,16),                                     : RT
002880 "Use",                                     : RT
002890      AT (21,20), FAC(HEX(85)), TAB$          , CH(03),      : RT
002900      AT (21,24),                                     : RT
002910 "or",                                     : RT
002920      AT (21,27), FAC(HEX(85)), BACKTAB$      , CH(07),      : RT
002930      AT (21,35),                                     : RT
002940 "to move between modifiable fields.",      : RT
002950      AT (23,30),                                     : RT
002960 "Press",                                     : RT
002970      AT (23,36), FAC(HEX(85)), ENTER$       , CH(05),      : RT
002980      AT (23,42),                                     : RT
002990 "to continue",                             : JM
003000      AT (24,13),                                     : JM
003010 "Press",                                     : RT
003020      AT (24,19), FAC(HEX(84)), PF16$        , CH(04),      : JM
003030      AT (24,24),                                     : JM
003040 "to exit without ",                         : JM
003041      AT (24,40), FAC(HEX(8C)), MESING$,      : JM
003050 KEYS(BIN(0)&BIN(16)&BIN(32)),                : RT
003060 ON (BIN(16)&BIN(32)) GOTO EXIOM,EXIOM        : RT
003070                                     : RT
003080 ER$=" ": E$=HEX(8C)                          : RT
003090 /* Check for validity of entries */          : RT
003100                                     : RT
003110      IF POS("BC"=IDFL$)>0 THEN NEXT2          : RT
003120      ER$ = " Identification flag is illegal." : RT
003130      E$ = HEX(94)                                : RT
003140      GOTO AOK                                     : RT
003150                                     : RT
003160 NEXT2: IF DEF$(1)<>" " THEN UPDATE          : RT
003170      ER$ = "Please enter your definition starting on line 1." : RT
003180      E$ = HEX(94)                                : RT
003190      GOTO AOK                                     : RT
003200                                     : RT
003210 /* EXIOM is used to abnormally exit the insert mode and */ : RT
003220 /* NOCHAN is used to abnormally exit the modify mode */   : RT
003230 EXIOM: READ #1, HOLD,KEY=WORD$,IOBUFF$        : RT
003240      IF NF=1 THEN NOCHAN                         : RT
003250      DELETE #1                                    : RT
003260      ER$= "                                     "&WORD$&" was not inserted." : RT
003270      GOTO CWMENU                                  : RT
003280                                     : RT
003290 NOCHAN: PROTECT$ = " ":P%=0                    : RT
003300      $PACK (P=F2$) IOBUFF$ FROM PROTECT$        : RT
003310      REWRITE #1,IOBUFF$,DATA GOTO IOERR2        : RT
003320      ER$= "                                     "No changes were saved." : RT
003330      ER1$=" "                                     : RT
003340      GOTO CWMENU                                  : RT
003350                                     : RT
003360 /* UPDATE removes the protection on the record and writes out */ : RT
003370 /* the final data as input. */                   : JM
003380 UPDATE: READ #1, HOLD, KEY=WORD$, IOBUFF$      : RT
003390      PROTECT$ = " ":P%=0                        : RT
003400      $PACK (P=F1$) IOBUFF$ FROM WORD$,IDFL$,DEF$(1),DEF$(2), : RT

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 11:25 11/09/84 Page 7  
Input File is CRESEBCG in Library CRESESRC on Volume CRES02

```

003410      DEF$(3),PROTECT$      RT
003420      REWRITE #1,IOBUFF$,DATA GOTO IOERR2      RT
003430      ER1$= " "      RT
003440      IF NF=0 THEN RITE      /* NF=0 indicates insert mode */      RT
003450      IF DF=1 THEN EXDEL      /* DF=1 indicates delete mode */      RT
003460      /* otherwise in modif/ mode */      RT
003470 ER$="""&WORD$&"" modifications updated - Please continue."      RT
003480      ES=HEX(8C)      RT
003490      WORD$=" "      RT
003500      GOTO CWMENU      RT
003510 RITE:      RT
003520      ER$= " ""&WORD$&"" inserted - Please continue."      RT
003530      ES= HEX(8C)      RT
003540      WORD$=" "      RT
003550      GOTO CWMENU      RT
003560      RT
003570 EXDEL: DF=0      RT
003580      X$=HEX(81)      JM
003590      PASSWRD$=HEX(000102030E101112131E20)      JM
003600      ER$ ="      Deletion cancelled."      RT
003610      GOTO CWMENU      RT
003620      RT
003630 EXIT: CLOSE #1      RT
003650      /* RETURN TO ABBREVIATION/ACRONYM GLOSSARY MENU */      JM

```

Technical Report 86-001

THIS PAGE INTENTIONALLY LEFT BLANK.

Create/Maintain Technical Glossary (CRESGLOS)

CRESGLOS in library CRES@SRC is the BASIC language program to create/maintain the Technical Glossary. The object language version is CRESGLOS in library CRES@SAV.

This program first allows the user to specify the Technical Glossary to be created or maintained. Then, the program enables the user to add, delete, or modify any given term in the glossary. There is a protection system implemented, and provision is made for clearing the protect bytes, if for some reason the bytes have been left set incorrectly.

Each entry in the Technical Glossary can contain up to 20 characters, identification of up to three Technical Word Lists on which it appears, a flag indicating whether it is a "Mandatory Technical Word" (one which is so basic to the particular technical area in which it is used that it is counted as a single syllable in reading grade level determinations), and a definition. The data record description is provided in appendix A.

A complete program listing follows. Appendix C contains a sample flowchart.

AD-A168 681

COMPUTER READABILITY AND EDITING SYSTEM (CRES)  
PROGRAMMER'S REFERENCE MANUAL(U) NAVAL TRAINING SYSTEMS  
CENTER ORLANDO FL S R SCHALOW ET AL. JAN 86

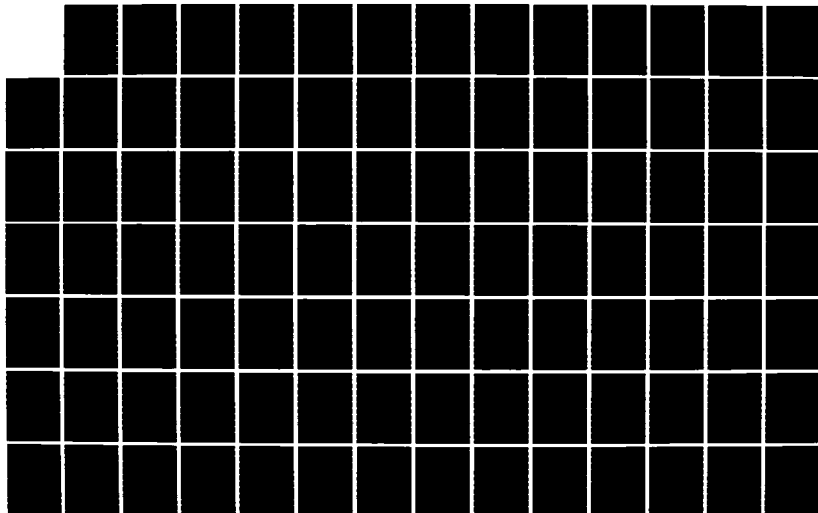
3/4

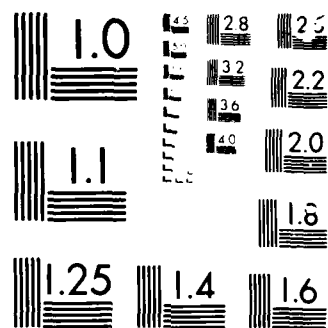
UNCLASSIFIED

NAVTRASYSCEN-TR-86-001

F/G 5/9

NL





MICROCOPY

CHART

Technical Report 86-001

THIS PAGE INTENTIONALLY LEFT BLANK.

## Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 12:50 10/30/84 Page 1  
Input file is CRES6LUS in library CRES0SRC on Volume CRES02

```

000010 /* CRES GENERAL GLOSSARY SUBSYSTEM PROGRAM */ RT
000020 /* */ RT
000030 /* AUTHOR : Rose Tank */ JM
000040 /* Date: March 12, 1984 */ RT
000050 /* */ RT
000060 /* Abstract: This program is used to create and/or maintain */ JM
000061 /* the CRES Glossary of Technical Words. */ JM
000070 RT
000080 DIM WORD$20, CODE$1, ER$65, PROLCIT$3, MTWFL$1 RT
000090 DIM F1$20, IOBUFF$215, ER1$50, F2$10, ILIB$8, IVOL$6 RT
000100 DIM FIL$(3)8, USER$3, USERNAME$24, WRKST$3 RT
000110 DIM HEADER$40, PF16$4 RT
000120 DIM ENTER$5, X$1, MESING$14 JM
000130 DIM DEF$(3)55, PASSWRD$11 RT
000140 DIM MODE$15, CURRENT ATTR$18, TAB$3, BACKTAB$7 RT
000150 CURRENT ATTR$="CURRENT ATTRIBUTES" RT
000160 TAB$="TAB": BACKTAB$="BACKTAB" RT
000170 ENTER$="ENTER" RT
000180 PF16$="PF16" RT
000190 PF$="PF": OPER$="- Operation" RT
000200 U$="1": T$="2": TH$="3": F1$="14": ST$="16" RT
000210 WELCOME$="Welcome to the" RT
000220 HEADER$="CRES GLOSSARY MAINTENANCE SUBSYSTEM" RT
000230 RT
000240 /* The following two lines are used as formats in the packing */ RT
000250 /* and the unpacking of record field values */ RT
000260 F1$=HEX(A014A008A0BAA0BAA037A017A037A001A003000?) RT
000270 F2$=HEX(0002A003000?) RT
000280 RT
000290 CALL "EXTRACT" ADDR("ID",USER$,"IL",ILIB$,"IV",IVOL$) RT
000300 RT
000310 /* SET UP THE USER FILE BLOCK */ RT
000320 RT
000330 SELECT #1,"GLOSSY",VARC,INDEXED,RECSIZE=215,KEYPOS=1,KEYLEN=20 RT
000340 RT
000350 CALL "LIMITS"("GLOSSARY",ILIB$,IVOL$,CODE$) RT
000360 IF CODE$="Y" THEN 430 RT
000370 OPEN NOGET Parm #1, OUTPUT, FILE="GLOSSARY", LIBRARY=ILIB$, : RT
000380 VOLUME=IVOL$ RT
000390 IOBUFF$=" CRES GLOSSARY" RT
000400 WRITE #1, IOBUFF$ RT
000410 CLOSE #1 RT
000420 RT
000430 OPEN NOGET Parm #1, SHARED, FILE="GLOSSARY", LIBRARY=ILIB$, : RT
000440 VOLUME=IVOL$ RT
000450 RT
000460 /* This section of code generates the first screen, which asks */ JM
000470 /* for the particular word and its corresponding operation - */ JM
000480 /* either insert word, delete word, or modify word. The user is */ RT
000490 /* also at this point given the option of returning to the */ JM
000500 /* Glossary Maintenance Menu. */ JM
000510 RT
000520 WORD$, ER$, ER1$=" " : F1$=HEX(8C) RT
000530 X$=HEX(81) JM
000540 PASSWRD$=HEX(000102030E101112131E20) JM
000550 RT
000560 CWMENU: RT

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 12:50 10/30/84 Page 2  
Input File is CRESGLOS in Library CRES@SRC on Volume CRES02

```

000550 ACCEPT : RT
000560 AT (01,03), : RT
000570 "CRES: Glossary Maintenance Subsystem (Word & Option Sel : RT
000580 ection Screen)", : RT
000590 AT (02,03), : RT
000600 "===== : RT
000610 =====", : JM
000620 AT (04,15), FAC(HEX(8C)), : RT
000630 WELCOME$ : RT
000640 AT (04,31), FAC(HEX(85)), : RT
000650 HEADER$, : RT
000660 AT (06,05), : RT
000670 "Please select a word:", : RT
000680 AT (08,32), FAC(X$), WORD$, , CH(20), : JM
000690 AT (10,18), FAC(E$), ER$, , CH(60), : RT
000700 AT (11,20), FAC(E$), ER1$, , CH(50), : RT
000710 AT (13,05), : RT
000720 "To continue, select the PF key corresponding to the desired oper : RT
000730 ation.", : RT
000740 AT (16,28), FAC(HEX(A5)), PF$, , CH(02), : RT
000750 AT (16,35), FAC(HEX(AC)), OPER$, , CH(31), : RT
000760 AT (18,28), FAC(HEX(84)), O$, , CH(01), : RT
000770 AT (18,35), : RT
000780 "INSERT a technical word", : RT
000790 AT (19,28), FAC(HEX(84)), T$, , CH(1), : RT
000800 AT (19,35), : RT
000810 "DELETE a technical word", : RT
000820 AT (20,28), FAC(HEX(84)), TH$, , CH(1), : RT
000830 AT (20,35), : RT
000840 "MODIFY a technical word", : RT
000850 AT (22,28), FAC(HEX(84)), FT$, , CH(02), : RT
000860 AT (22,35), : RT
000870 "Protection Override", : RT
000880 AT (23,28), FAC(HEX(84)), ST$, , CH(02), : RT
000890 AT (23,35), : RT
000900 "Return to the Glossary Maintenance Menu", : RT
000910 KEYS(PASSWRD$), : JM
000920 KEY(OPTION) : JM
000930 IF OPTION=16 OR OPTION=32 THEN EXIT : JM
000940 : RT
000950 WELCOME$,HEADER$=" " : RT
000960 ER$,ER1$=" ": E$=HEX(8C) : RT
000970 : RT
000980 /* Check for confirmation of deletion */ : RT
000990 IF (OPTION=0)AND(DF=1) THEN DEL : RT
001000 /* Check for cancellation of deletion */ : JM
001010 IF DF=1 THEN UPDATE : RT
001020 /* Check for override of protection byte */ : RT
001030 IF (OPTION=14) OR (OPTION=30) THEN OVERRIDE : RT
001040 IF STR(WORD$,1,1)<>" " THEN REED : RT
001050 : RT
001060 PRINT BELL. : RT
001070 ER$=" The first character cannot be blank." : RT
001080 ER1$=" Please reenter the appropriate word and PF key." : RT
001090 E$= HEX(94) : RT
001100 GOTO CWMENU : RT
001110 : RT

```

[illegible]

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 12:50 10/30/84 Page 4  
Input File is CRESGLOS in Library CRES0SRC on Volume CRES02

```

001680          GOTO CWMENU                                RT
001690                                                    RT
001700 /* Set the protection byte */                        RT
001710 NOW:      PROTECT$= USER$                            RT
001720          $PACK (F=F1$) IOBUFF$ FROM WORD$,FIL$(1),FIL$(2),FIL$(3): RT
001730          ,DEF$(1),DEF$(2),DEF$(3),MTWFL$,PROTECT$      RT
001740          REWRITE #1, IOBUFF$, DATA GOTO IOERR2          RT
001750                                                    RT
001760          /* NOW BRANCH TO THE OPTION SELECTED */        RT
001770 OPT:      ON OPTION GOTO INSERT,DELETE,MODIFY          RT
001780          ON OPTION-16 GOTO INSERT,DELETE,MODIFY          RT
001790                                                    RT
001800 NOTFOUND: NF=0 /* The word was not found in the list */ RT
001810          GOTO OPT                                        RT
001820                                                    RT
001830 INSERT:                                              RT
001840          /* NF=0 indicates the word doesn't yet exist so go ahead */ RT
001850          /* and insert. NF=1 indicates the word exists. */ RT
001860          IF NF=1 THEN EXIST                              RT
001870          MODE$= "(Insert Screen)"                        RT
001871          MESING$="inserting word"                        JM
001880          FIL$(1),DEF$(1),MTWFL$=" ":P%=0                RT
001890          IOBUFF$=" ": PROTECT$= USER$                    RT
001900          $PACK (F=F1$) IOBUFF$ FROM WORD$,FIL$(1),FIL$(2),FIL$(3),: RT
001910          DEF$(1),DEF$(2),DEF$(3),MTWFL$,PROTECT$        RT
001920          WRITE #1, IOBUFF$, DATA GOTO IOERR2            RT
001930          GOTO INS                                         RT
001940                                                    RT
001950 /* Coming from insert with the word currently existing, */ RT
001960 /* print an error message and erase the protection byte */ RT
001970 EXIST:                                              RT
001980          ER$= "The word '" & WORD$ & "' already exists in this list." RT
001990          E$= HEX(94)                                      RT
002000          READ #1, HOLD, KEY= WORD$,IOBUFF$              RT
002010          PROTECT$=" ":P%=0                               RT
002020          $PACK (F=F1$) IOBUFF$ FROM WORD$,FIL$(1),FIL$(2),FIL$(3), : RT
002030          DEF$(1),DEF$(2),DEF$(3),MTWFL$,PROTECT$        RT
002040          REWRITE #1, IOBUFF$, DATA GOTO IOERR2          RT
002050          GOTO CWMENU                                      RT
002060                                                    RT
002070 /* DELET,DEL,MODEL- all pertain to the delete option */ RT
002080 /* DELET- gives the user a chance to cancel the deletion */ RT
002090 DELET: IF NF=0 THEN MODEL                              RT
002100          DF=1                                              RT
002110          ER$= "      Press ENTER to confirm deletion of "& WORD$ RT
002120          ER1$="      or enter PF14 to cancel deletion."  RT
002121          X$=HEX(84)                                       JM
002122          PASSWRD$=HEX(000E1E)                             JM
002130          E$= HEX(94)                                      RT
002140          GOTO CWMENU                                      RT
002150                                                    RT
002160 /* DEL- deletes the entire record */                    RT
002170 DEL:      READ #1, HOLD,KEY=WORD$,IOBUFF$              RT
002180          DELETE #1                                         RT
002190          DF=0                                              RT
002200          ER$= "      '"& WORD$&"' deleted - Please continue." JM
002210          E$= HEX(8C)                                       JM

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 12:50 10/30/84 Page 5  
Input File is CRESGLOS in Library CRES@SRC on Volume CRES02

```

002211      X$=HEX(81)                                JM
002212      PASSWRD$=HEX(000102030E101112131E20)      JM
002220      WORD$=" "                                  RT
002221      NF=0                                         RT
002230      GOTO CWMENU                                RT
002240                                             RT
002250      /* NODEL- print an error message when the word doesn't exist */ RT
002260      NODEL: ER$= WORD$ & "' does not exist in this list." RT
002270      E$= HEX(94)                                  RT
002280      GOTO CWMENU                                RT
002290                                             RT
002300      MODIFY: IF NF=1 THEN GETIT                    RT
002310                                             /* if word not in list, */ RT
002320      ER$= " Modification request denied -"          RT
002330      ER1$=WORD$ & " does not exist in this list." RT
002340      E$=HEX(94) /* set up message and fac and */ RT
002350      GOTO CWMENU /* go back up to previous screen */ RT
002360                                             RT
002370      GETIT:/*READ #1,HOLD,KEY=WORD$,IOBUFF$,EOD GOTO NOTFOUND, */ RT
002380      /*DATA GOTO IOERR2 */ RT
002390                                             RT
002400      MODE$      =(Modify Screen)" RT
002405      MESING$   ="modifying word" JM
002410      INS:                                             RT
002420                                             RT
002430      /* This accept statement generates a screen with the attributes */ RT
002440      /* corresponding to each word appropriately labeled for insert */ RT
002450      /* or modification. */ RT
002460                                             RT
002470      AOK:                                             RT
002480                                             RT
002490      ACCEPT : RT
002500      AT (01,03), : RT
002510      "CRES: Glossary Subsystem ", AT(01,66),FAC(HEX(8C)),MODE$, : RT
002520      AT (02,03), : RT
002530      "===== : RT
002540      =====, : JM
002550      AT (04,03), : RT
002560      " Please enter the appropriate attributes.", : RT
002570      AT (07,05), : RT
002580      "WORD:", : RT
002590      AT (07,14), FAC(HEX(85)), WORD$ , CH(18), : RT
002600      AT (10,05), : RT
002610      "APPEARING ON WORD LIST(S): 1)", : RT
002620      AT (10,36), FAC(HEX(81)), FIL$(1) , CH(08), : RT
002630      AT (10,48), : RT
002640      "2)", : RT
002650      AT (10,51), FAC(HEX(81)), FIL$(2) , CH(08), : RT
002660      AT (10,63), : RT
002670      "3)", : RT
002680      AT (10,66), FAC(HEX(81)), FIL$(3) , CH(08), : RT
002690      AT (12,05), : RT
002700      "DEFINITION:", : RT
002710      AT (12,19), FAC(HEX(81)), DEF$(1) , CH(55), : RT
002720      AT (13,19), FAC(HEX(81)), DEF$(2) , CH(55), : RT
002730      AT (14,19), FAC(HEX(81)), DEF$(3) , CH(55), : RT
002740      AT (16,05), : RT

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 12:50 10/30/84 Page 6  
Input File is CRESGLOS in Library CRES0SRC on Volume CRES02

```

002750 "MANDATORY TECHNICAL WORD:", RT
002760 AT (16,36), FAC(HEX(81)), MTWFL$ , CH(01), RT
002770 AT (16,44), RT
002780 "(Legal entries are ' ' or '*')", RT
002790 AT (18,15), FAC(E$), ER$ , CH(55), RT
002800 AT (20,15), RT
002810 "Use", RT
002820 AT (20,19), FAC(HEX(85)), TAB$ , CH(03), RT
002830 AT (20,23), RT
002840 "or", RT
002850 AT (20,26), FAC(HEX(85)), BACKTAB$ , CH(07), RT
002860 AT (20,34), RT
002870 "to move between modifiable fields.", RT
002880 AT (23,30), RT
002890 "Press", RT
002900 AT (23,36), FAC(HEX(85)), ENTER$ , CH(05), RT
002910 AT (23,42), RT
002920 "to continue", RT
002930 AT (24,21), JM
002940 "Press", RT
002950 AT (24,27), FAC(HEX(84)), PF16$ , CH(04), JM
002960 AT (24,32), JM
002970 "to exit without ", JM
002971 AT (24,48), FAC(HEX(8C)), MESING$, JM
002980 KEYS(BIN(0)&BIN(16)&BIN(32)), RT
002990 ON (BIN(16)&BIN(32)) GOTO EXIOM,EXIOM RT
003000 RT
003010 ER$=" ": E$=HEX(8C) RT
003020 /* Check for validity of entries */ RT
003030 RT
003040 /* Make sure the files are stored consecutively */ RT
003050 FL%=0 RT
003060 FOR I= 1 TO 2 RT
003070 IF FIL$(I)<>" " OR FIL$(I+1)=" " THEN 3090 RT
003080 FIL$(I)=FIL$(I+1):FIL$(I+1)=" ":FL%=1 RT
003090 NEXT I RT
003100 IF FL%=1 THEN 3050 RT
003110 RT
003120 /* Make sure the definitions are stored consecutively */ RT
003121 FL%=0 RT
003122 FOR I= 1 TO 2 RT
003123 IF DEF$(I)<>" " OR DEF$(I+1)=" " THEN 3125 RT
003124 DEF$(I)=DEF$(I+1):DEF$(I+1)=" ":FL%=1 RT
003125 NEXT I RT
003126 IF FL%=1 THEN 3121 RT
003130 RT
003140 IF POS(" * "=MTWFL$)>0 THEN 3190 RT
003150 ER$=" Mandatory technical word flag value is illegal." RT
003160 E$= HEX(94) RT
003170 GOTO AOK RT
003180 RT
003190 IF DEF$(I)<>" " THEN UPDATE RT
003200 ER$ = "Please enter your definition starting on line 1." RT
003210 E$ = HEX(94) RT
003220 GOTO AOK RT
003230 RT
003240 /* EXIOM is used to abnormally exit the insert mode and */ RT

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 12:50 10/30/84 Page 7  
Input File is CRESGLOS in Library CRES0SRC on Volume CRES02

```

003250 /* NOCHAN is used to abnormally exit the modify mode */ RT
003260 EXIOM: READ #1, HOLD, KEY=WORD$, IOBUFF$ RT
003270 IF NF=1 THEN NOCHAN RT
003280 DELETE #1 RT
003290 ER$= " " &WORD$ " was not inserted." RT
003300 E$= HEX(84) RT
003310 GOTO CWMENU JM
003320 JM
003330 NOCHAN: PROTECT$ = " " JM
003340 $PACK (F=F2$) IOBUFF$ FROM PROTECT$ JM
003350 REWRITE #1, IOBUFF$, DATA GOTO IOERR2 JM
003360 ER$= " " No changes were saved." JM
003370 ER1$= " " :E$=HEX(84) JM
003380 GOTO CWMENU JM
003390 JM
003400 /* UPDATE removes the protection on the record and writes out */ JM
003410 /* the final data as input. */ JM
003420 UPDATE: READ #1, HOLD, KEY=WORD$, IOBUFF$ JM
003430 PROTECT$ = " " :P%=0 JM
003440 $PACK (F=F1$) IOBUFF$ FROM WORD$, FIL$(1), FIL$(2), FIL$(3), : JM
003450 DEF$(1), DEF$(2), DEF$(3), MTWFL$, PROTECT$ JM
003460 REWRITE #1, IOBUFF$, DATA GOTO IOERR2 JM
003470 ER1$= " " JM
003480 IF NF=0 THEN RITE /* NF=0 indicates insert mode */ JM
003490 IF DF=1 THEN EXDEL /* DF=1 indicates delete mode */ JM
003500 /* otherwise in modify mode */ JM
003510 ER$="" &WORD$ " modifications updated - Please continue." JM
003520 E$=HEX(8C) JM
003530 WORD$="" JM
003540 GOTO CWMENU JM
003550 RITE: JM
003560 ER$= " " &WORD$ " inserted - Please continue." JM
003570 E$= HEX(8C) JM
003580 WORD$="" JM
003590 GOTO CWMENU JM
003600 JM
003610 EXDEL: DF=0 JM
003620 ER$="" Deletion cancelled." JM
003621 X$=HEX(81) JM
003622 PASSWRD$=HEX(000102030E101112131E20) JM
003630 GOTO CWMENU JM
003640 JM
003650 EXIT: CLOSE #1 JM

```

Technical Report 86-001

THIS PAGE INTENTIONALLY LEFT BLANK.

Print Abbreviation/Acronym Glossary (CRES@BCP)

CRES@BCP in library CRES@SRC is the BASIC language program to print the Abbreviation/Acronym Glossary. The object language version is CRES@BCP in library CRES@SAV.

This program first allows the user to specify the name of the Abbreviation/Acronym Glossary to be printed. Then the program enables the user to print out either the list of abbreviations, the list of acronyms, or the combined list. The glossary can be printed out in its entirety or some specified range can be printed. This printout can be displayed at the terminal or sent to the printer. An optional subtitle of up to two lines can be included for the printout, and the number of lines per page may be specified as some value less than the default value of 50.

A complete program listing follows. Appendix C contains a sample flowchart.

Technical Report 86-001

THIS PAGE INTENTIONALLY LEFT BLANK.

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 13:22 10/30/84 Page 1  
Input File is CRES@BCP in Library CRES@SRC on Volume CRES02

```

000010 /* CRES ABBREVIATION/ACRONYM GLOSSARY PRINT PROGRAM      */ RT
000020 /*                                                              */ RT
000030 /*              AUTHOR: Rose Tank                             */ JM
000040 /*              Date:   March 12, 1984                         */ RT
000050 /*                                                              */ RT
000060 /*                                                              */ RT
000070 /* Abstract: This program is used to print or display all or   */ JM
000080 /* part of the Abbreviation/Acronym List.                      */ JM
000130 /*                                                              */ RT
000140 DIM CODE$1,MSG$53,MSG$255,MSG2$40,RLIB$8                      RT
000150 DIM IOBUFF$190,SLIB$8,OPT$19                                RT
000160 DIM FILE$08,LIB$08,SVOL$6                                    RT
000170 DIM IVOL$06,HEADER$51                                       RT
000180 DIM SUBTIT1$20,SUBTIT2$20                                   RT
000190 DIM ENTER$          05,RVOL$6                               RT
000200 DIM ER$              58                                       RT
000210 DIM TOP$            08                                       RT
000220 DIM PF15$           04                                       RT
000230 DIM PF16$           04                                       RT
000240 DIM X$              01                                       RT
000250 DIM A$(3)          01                                       RT
000260 DIM START$          15                                       RT
000270 DIM END$            15                                       RT
000280 /*                                                              */ RT
000290 ENTER$="ENTER"                                              RT
000300 PF15$="PF15"                                                RT
000310 PF16$="PF16"                                                RT
000320 WELCOME$="Welcome to the "                                  RT
000330 HEADER$="CRES ABBREVIATION/ACRONYM PRINT SUBSYSTEM"      RT
000340 FILE$="ABBRELST"                                           RT
000350 CALL "EXTRACT" ADDR("IL",LIB$,"IV",IVOL$,"SL",SLIB$,"SV",SVOL$, :
      "RL",RLIB$,"RV",RVOL$)                                     RT
000360 /*                                                              */ RT
000370 SELECT #1,"ABBREV",VARC,INDEXED,RECSIZE=190,KEYPOS=1,KEYLEN=20, : RT
000380 IOERR GOTO IOERR1                                           RT
000390 /*                                                              */ RT
000400 SELECT #2,"PRTFIL",PRINTER,RECSIZE=120, IOERR GOTO IOERR2   RT
000410 /*                                                              */ RT
000420 OPEN NOGETPARM #1,SHARED, FILE= FILE$, LIBRARY = LIB$, : RT
000430 VOLUME = IVOL$                                             RT
000440 /*                                                              */ RT
000450 /* This section of code generates the first screen, which asks */ JM
000460 /* for the particular word interval and its corresponding      */ RT
000470 /* print operation. The user is given the option of returning */ RT
000480 /* to the Glossary Maintenance Menu, if desired.              */ JM
000510 /*                                                              */ RT
000520 E$=HEX(8C)                                                  RT
000530 X$="X"                                                    RT
000540 LPP%=50                                                    RT
000550 ENTER$="ENTER"                                           RT
000560 PF16$="PF16"                                             RT
000570 PRINTLN:                                                 RT
000580 START$="FIRST"                                           RT
000590 END$="LAST"                                             RT
000600 TOP$="TERMINAL"                                           RT
000610 A$(),SUBTIT1$, SUBTIT2$=""                               RT
000620 PRTS$CREEN:                                             RT
000630 ACCEPT                                                  : RT

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 13:22 10/30/84 Page 2  
Input File is CRES@BCP in Library CRES@SRC on Volume CRES02

```

000640      AT (01,03), RT
000650 "CRES: Abbreviation/Acronym Print Subsystem (List Specific RT
000660 ation Screen)", RT
000670      AT (02,03), RT
000680 "===== RT
000690 "===== RT
000700 AT(04,10), FAC(HEX(8C)), WELCOME$,AT(04,25),FAC(HEX(85)),HEADER$, RT
000710      AT (06,03), RT
000720 " Please place an", RT
000730      AT (06,21), FAC(HEX(85)), X$ , CH(01), RT
000740      AT (06,23), RT
000750 "next to the appropriate box to specify which list you", RT
000760      AT (07,03), RT
000770 " want to print.", RT
000780      AT (08,34), RT
000790 "Abbreviations", RT
000800      AT (08,50), FAC(HEX(81)), A$(1) , CH(01), RT
000810      AT (09,34), RT
000820 "Acronyms", RT
000830      AT (09,50), FAC(HEX(81)), A$(2) , CH(01), RT
000840      AT (10,34), RT
000850 "Both", RT
000860      AT (10,50), FAC(HEX(81)), A$(3) , CH(01), RT
000870      AT (12,13), FAC(E$), ER$ , CH(58), RT
000880      AT (14,03), RT
000890 " Please supply the following information for the print system:" RT
000900      AT (15,25), RT
000910 "Subtitle (optional)", RT
000920      AT (16,54), FAC(HEX(81)), SUBTIT1$ , CH(20), RT
000930      AT (17,54), FAC(HEX(81)), SUBTIT2$ , CH(20), RT
000940      AT (18,25), RT
000950 "Starting abbrev/acronym", RT
000960      AT (18,54), FAC(HEX(81)), START$ , CH(15), RT
000970      AT (19,25), RT
000980 "Ending abbrev/acronym", RT
000990      AT (19,54), FAC(HEX(81)), END$ , CH(15), RT
001000      AT (20,25), RT
001010 "Lines per page", RT
001020      AT (20,54), LPP% , PIC(##), RT
001030      AT (21,25), RT
001040 "Print at (TERMINAL/PRINTER)", RT
001050      AT (21,54), FAC(HEX(81)), TOP$ , CH(08), RT
001060      AT (23,30), RT
001070 "Press", RT
001080      AT (23,36), FAC(HEX(85)), ENTER$ , CH(05), RT
001090      AT (23,42), RT
001100 "to continue ", RT
001110      AT (24,15), RT
001120 "Press", RT
001130      AT (24,21), FAC(HEX(84)), PF16$ , CH(04), RT
001140      AT (24,26), RT
001150 "to return to the Glossary Maintenance Menu ", RT
001160 KEYS(BIN(16)&BIN(32)&BIN(0)), RT
001170 ON (BIN(16)&BIN(32)) GOTO EXIT,EXIT RT
001180 RT
001190 ER$=" ":E$=HEX(8C):WELCOME$,HEADER$=" " RT
001200 IF A$(1)=" " AND A$(1)=A$(2) AND A$(3)="X" THEN 1260 RT

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 13:22 10/30/84 Page 3  
Input File is CRES0BCP in Library CRES0SRC on Volume CRES02

```

001210 IF A$(2)=" " AND A$(2)=A$(3) AND A$(1)="X" THEN 1260 RT
001220 IF A$(3)=" " AND A$(1)=A$(3) AND A$(2)="X" THEN 1260 RT
001230 ER$= " Please mark only one category with an 'X'" RT
001240 E$=HEX(94) RT
001250 GOTO PRGSCREEN RT
001260 IF STR(TOP$,1,1) = "P" THEN NEXT11 RT
001270 IF STR(TOP$,1,1) <> "T" THEN TERMPR RT
001280 NEXT11: IF LPP% < 51 THEN NEXT13 RT
001290 ER$= " 'Lines per page' can be at most 50." RT
001300 E$=HEX(84) RT
001310 GOTO PRGSCREEN RT
001320 IOERR1: PRINT PAGE RT
001330 PRINT BELL, HEX(94); "ERROR in ABBREV/ACRONYM list." RT
001340 FOR I=1 TO 500000 :NEXT I RT
001350 CLOSE #1 RT
001360 GOTO PRGSCREEN RT
001370 IOERR2: PRINT PAGE RT
001380 PRINT BELL, HEX(94); "ERROR in print file." RT
001390 FOR I=1 TO 500000 :NEXT I RT
001400 CLOSE #2 RT
001410 GOTO PRGSCREEN RT
001420 TERMPR: E$=HEX(94) RT
001430 ER$="Print at ' has only legal values TERMINAL or PRINTER." RT
001440 GOTO PRGSCREEN RT
001450 RT
001460 REX: ER$= " 'Starting point' comes after 'Ending point'." RT
001470 E$=HEX(94):GOTO PRGSCREEN RT
001480 RT
001490 RT
001500 NEXT13: IF START$="FIRST" OR END$="LAST" THEN FORD JM
001510 IF START$>END$ THEN REX JM
001520 RT
001530 /* Now it's time to generate our print file name */ RT
001540 RT
001550 FORD: DAT$= DATE JM
001560 D$= STR(DAT$,3,2)&"/"&STR(DAT$,5,2)&"/"&STR(DAT$,1,2) RT
001570 T$= STR(TIME,1,2)&":"&STR(TIME,3,2) RT
001580 RT
001590 DISPLAY : RT
001600 AT (01,03), : RT
001610 "CRES: Abbreviation/Acronym Glossary Print Subsystem", : RT
001620 AT (01,61), : RT
001630 "(Generation Screen)", : RT
001640 AT (02,03), : RT
001650 "===== : RT
001660 =====", : RT
001670 AT (12,23), : RT
001680 "Output file now being generated!" RT
001690 PFILE$="ABBREV##" RT
001700 I%=0 RT
001710 NEXT14: CONVERT I% TO STR(PFILE$,7,2),PIC(00) RT
001720 I%=I%+1 RT
001730 CALL "LIMITS" (PFILE$,SLIB$,SVOL$,CODE$,MSG$) RT
001740 IF CODE$="Y" THEN NEXT14 RT
001750 OPEN NOGETPARM #2,OUTPUT,FILE=PFILE$,LIBRARY=SLIB$, : RT
001760 VOLUME=SVOL$,SPACE=8000 RT
001770 GOSUB TTL_PGE RT

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 13:22 10/30/84 Page 4  
Input File is CRES@BCP in Library CRES@SRC on Volume CRES02

```

001780      IF START$ <> "FIRST" THEN 1800      RT
001790      START$ = " BB"      RT
001800      IF END$ = "LAST" THEN END$ = ALL(HEX(FF))      RT
001810      PAGE% = 1      RT
001820      RECS% = 1      RT
001830      READ #1, KEY$=START$,IOBUFF$,EOD GOTO HOME      JM
001840      GOSUB HEADER      RT
001850      IF STR(IOBUFF$,20)>END$ THEN HOME      RT
001860      KNOT:      RT
001870      IF A$(1)="X" AND STR(IOBUFF$,21,1)="B" THEN 1910      RT
001880      IF A$(2)="X" AND STR(IOBUFF$,21,1)="C" THEN 1910      RT
001890      IF A$(3)="X" THEN 1910      RT
001900      GOTO AGAN      RT
001910      WRITE #2, USING IMG6,C$,STR(IOBUFF$,1,20),      ! RT
001920      STR(IOBUFF$,21,1),STR(IOBUFF$,22,55)      RT
001930      RECS%=RECS%+1      RT
001940      IF STR(IOBUFF$,77,55)=" " THEN 1970      RT
001950      WRITE #2, USING IMG5,C$,STR(IOBUFF$,77,55)      RT
001960      RECS%=RECS%+1      RT
001970      IF STR(IOBUFF$,132,55)=" " THEN 2000      RT
001980      WRITE #2, USING IMG5,C$,STR(IOBUFF$,132,55)      RT
001990      RECS%=RECS%+1      RT
002000      IF RECS%>LPP% THEN RECS%=1      RI
002010      IF RECS%=1 THEN PAGE%=PAGE%+1      RT
002020      IF RECS% = 1 THEN GOSUB HEADER      RT
002030      AGAN: READ #1, IOBUFF$,EOD GOTO HOME      RT
002040      IF STR(IOBUFF$,20)>END$ THEN HOME      RT
002050      GOTO KNOT      RT
002060      HOME: WRITE #2, USING TIT, C$      RT
002070      WRITE #2, USING TIT, C$      RT
002080      WRITE #2, USING IMG8, C$      RT
002090      CLOSE#2      RT
002100      IF STR(TOP$,1,1) ="P" THEN MESS      RT
002110      CLOSE WS      RT
002120      MSG2$ = "CANCEL Processing"      RT
002130      MSK$=HEX(00)      RT
002140      CMSG$ = " "      RT
002150      CALL "LINK" ADDR("DISPFILE","P",RLIB$,RVOL$,3%,PFILE$,      ! RT
002160      SLIB$,SVOL$, "C",MSG2$,27%, "H",MSK$,CMSG$,255%,C%,R%)      RT
002170      ER$ = " " :E$=HEX(RC)      RT
002180      IF C%<0 OR R%<0      ! RT
002190      THEN ER$ = STR(CMSG$,141)      RT
002200      GOTO PRINTSCRN      RT
002210      RT
002220      MESS: ER$="File "&PFILE$&" in Library "& SLIB$ &" on "& SVOL$ & : RT
002230      " generated.":E$=HEX(84)      RT
002240      GOTO PRINTSCRN      RT
002250      RT
002260      TIT_PGE:      RT
002270      C$=HEX(8001)      RT
002280      WRITE #2, USING TIT, C$      RT
002290      C$ = HEX(0015)      RT
002300      WRITE #2, USING TIT, C$      RT
002310      C$ = HEX(0001)      RT
002320      WRITE #2, USING TIT1, C$      RT
002330      WRITE #2, USING TIT2, C$      RT
002340      WRITE #2, USING TIT0, C$,D$,T$      RT

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 13:22 10/30/84 Page 5  
Input File is CRES0BCP in Library CRES0SRC on Volume CRES02

```

002350      WRITE #2, USING TIT3, C$      RT
002360      WRITE #2, USING TIT2, C$      RT
002370      WRITE #2, USING TIT1, C$      RT
002380      WRITE #2, USING TIT2, C$      RT
002390      IF SUBTIT1$ = "                " THEN GERRY      RT
002400      WRITE #2, USING TIT4, C$, SUBTIT1$      RT
002410      IF SUBTIT2$ <> "                " THEN WRITE #2, USING : RT
002420 TIT10, C$, SUBTIT2$      RT
002430      WRITE #2, USING TIT2, C$      RT
002440 GERRY:      RT
002450      WRITE #2, USING TIT8, C$, START$      RT
002460      WRITE #2, USING TIT2, C$      RT
002470      WRITE #2, USING TIT9, C$, END$      RT
002480      WRITE #2, USING TIT2, C$      RT
002481      IF A$(1)="X" THEN OPT$="PRINT ABBREVIATIONS"
002486      IF A$(2)="X" THEN OPT$="PRINT ACRONYMS"
002488      IF A$(3)="X" THEN OPT$="PRINT BOTH"
002490      WRITE #2, USING TIT5, C$, OPT$      RT
002500      WRITE #2, USING TIT2, C$      RT
002510      WRITE #2, USING TIT1, C$      RT
002520      RETURN      RT
002530      RT
002540 HEADER:      RT
002550      IF PAGE%=1 THEN FIRST      RT
002560      C$ = HEX(0001)      RT
002570      WRITE #2, USING TIT, C$      RT
002580 FIRST: C$ = HEX(8001)      RT
002590      WRITE #2, USING TIT, C$      RT
002600      C$ = HEX(0001)      RT
002610      IF SUBTIT1$ = "                "      RT
002620      THEN WRITE #2, USING IMG,C$,D$,T$,PAGE%      RT
002630      ELSE GOTO FOUT      RT
002640      GOTO ALRITE      RT
002650 FOUT:  WRITE #2, USING IMG1,C$,SUBTIT1$,D$,T$,PAGE%      RT
002660      IF SUBTIT2$ <> "                "      RT
002670      THEN WRITE #2, USING IMG7,C$,SUBTIT2$      RT
002680 ALRITE: WRITE #2, USING IMG2, C$,LIB$,IVOL$      RT
002690      WRITE #2, USING IMG3, C$      RT
002700      WRITE #2, USING IMG4, C$      RT
002710      WRITE #2, USING IMG5, C$      RT
002720      WRITE #2, USING IMG3, C$      RT
002730      RETURN      RT
002740      RT
002750 TIT0:###      : #####      RT
002760 #####      :      RT
002770 TIT:###      :      RT
002780 TIT1:###      :-----:      RT
002790 -----:      RT
002800 TIT2:###      :      RT
002810      :      RT
002820 TIT3:###      : ***** ABBREVIATION/ACRONYM GLOSSARY: RT
002830 *****      :      RT
002840 TIT4:###      : TITLE OF LIST : #####! JM
002850 #####      :      RT
002860 TIT5:###      : OPTION SELECTED : #####! JM
002870 #####      :      RT
002880 TIT8:###      : STARTING POINT : #####! JM

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 13:22 10/30/84 Page 6  
Input File is CRES@BCP in Library CRES@SRC on Volume CRES02

```

002890 ##### : RT
002900 TIT9:## : ENDING POINT : #####! JM
002910 ##### : RT
002920 TIT10:## : #####! RT
002930 ##### : RT
002940 IMG:## ##### : RT
002950 PAGE ##### RT
002960 IMG1:## TITLE OF LIST : ##### : RT
002970 ##### PAGE ##### RT
002980 IMG7:## ##### RT
002990 IMG2:## Input File is ABBRELST in Library ##### on Volume ##! RT
003000 ## RT
003010 IMG3:## =====! RT
003020 ===== RT
003030 IMG4:## Abbrev/Acron ym Type Definition RT
003040 IMG6:## ##### # #####! RT
003050 ##### RT
003060 IMG5:## #####! RT
003070 ##### RT
003080 IMG8:## ;!! END OF LISTING !! RT
003090 RT
003100 EXIT: CLOSE #1 RT

```

Print Technical Glossary (CRESGLOP)

CRESGLOP in library CRES@SRC is the BASIC language program to print the Technical Glossary. The object language version is CRESGLOP in library CRES@SAV.

This program first allows the user to specify the name of the Technical Glossary to be printed. Then, the program enables the user to print out either only the Mandatory Technical Words that appear or the entire Technical Glossary. The glossary can be printed out in its entirety or some specified range can be printed. This printout can be displayed at the terminal or sent to the printer. An optional subtitle of up to two lines can be included for the printout, and the number of lines per page may be specified as some value less than the default value of 50.

A complete program listing follows. Appendix C contains a sample flowchart.

Technical Report 86-001

THIS PAGE INTENTIONALLY LEFT BLANK.

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 15:18 11/13/84 Page 1  
Input File is CRESGLOP in Library CRES@SRC on Volume CRES02

```

000010 /* CRES MANDATORY TECHNICAL GLOSSARY PRINT SUBSYSTEM PROGRAM */
000025 /* */
000040 /*          AUTHORS: Rose Tank and John Mears */
000055 /*          Date:   October 12, 1983 */
000070 /* */
000085
000100 /* This subsystem is not integrated with any other part of the */
000115 /* CRES subsystem. The glossaries are used exclusively for the */
000130 /* user's reference. The following code allows any portion of */
000145 /* the mandatory technical glossary to be printed for easy */
000160 /* viewing. */
000175 /* */
000190 /* Record format for the mandatory technical glossary: */
000205 /*          columns          contents */
000220 /*          1-20              word */
000235 /*          21-28              technical word list in which the word */
000250 /*                               is presently residing (optional) */
000265 /*          29-83              definition line 1 */
000280 /*          84-138             definition line 2 */
000295 /*          139-193            definition line 3 */
000310 /*          194-196            protection ID */
000325 /*          197-200            filler */
000340 /* */
000355 /* NOTE: Each of the definition lines consist of a 55 char. */
000370 /*          buffer. */
000385
000400
000415
000430 DIM CODE$1,IOBUFF$215,FILE$8,LIB$8,IVOL$6,ER$58,TOP$8,START$20
000445 DIM ENDS$20,HEADER$40,SUBTIT1$20,SUBTIT2$20
000460 DIM ENTER$5,PF15$4,PF16$4,X$1,RLIB$8,RVOL$6
000475 DIM MSG$53,CMSG$255,MSG2$40,SLIB$8,SVOL$6,OPT$20
000490
000505 ENTER$="ENTER":PF15$="PF15":PF16$="PF16"
000520 E$=HEX(8C):X$="X"
000535 WELCOME$="Welcome to the "
000550 HEADER$="CRES GLOSSARY PRINT SUBSYSTEM"
000552 CALL "EXTRACT" ADDR("IL",LIB$,"IV",IVOL$,"SL",SLIB$,"SV",SVOL$,
000554 "RL",RLIB$,"RV",RVOL$)
000565 FILE$="GLOSSARY"
000580
000595 SELECT #1,"GLOSSY",VARC,INDEXED,RECSIZE=215,KEYPOS=1,KEYLEN=20,
000610 IOERR GOTO IOERR1
000625
000640 SELECT #2,"PRTFIL",PRINTER,RECSIZE=120, IOERR GOTO IOERR2
000655
000670 OPEN NOGETPARM #1,SHARED, FILE= FILE$, LIBRARY = LIB$,
000685 VOLUME = IVOL$
000700
000715 /* This screen allows you to choose the range of words, the */
000730 /* number of lines per page, and the destination of the output */
000745 /* (terminal/printer). */
000760
000775 PRINTSCRN:
000790 START$="FIRST" /* The following are the defaults*/
000805 ENDS$="LAST" /* for the print screen */
000820 LPP$=50

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 15:18 11/13/84 Page 2  
Input File is CRESGLOP in Library CRES@SRC on Volume CRES02

```

000835 TOP$="TERMINAL"
000850 SUBTIT1$, SUBTIT2$ =" "
000865
000880 PRTSCREEN:
000895 ACCEPT
000910 AT (01,03),
000925 "GRES: Glossary Print Subsystem", RT
000940 AT (01,66),
000955 "(Print Screen)",
000970 AT (02,03),
000985 "=====
001000 =====",
001006 AT(04,15), FAC(HEX(8C)), WELCOME$,AT(04,30),FAC(HEX(85)),HEADER$, RT
001015 AT (06,02), RT
001030 " Please place an", RT
001045 AT (06,20), FAC(HEX(85)), X$ , CH(01), RT
001060 AT (06,22), RT
001075 "in the box below, if you want to print only the mandatory", RT
001090 AT (07,02), RT
001105 " technical words that appear in the glossary.", RT
001120 AT (09,25), RT
001135 "Mandatory Technical Word", RT
001150 AT (09,52), FAC(HEX(81)), AS , CH(01), RT
001165 AT (11,13), FAC(E$), ER$ , CH(58), RT
001180 AT (13,02), RT
001195 " Please supply the following information for the print system:" RT
001210 ,AT (15,25), RT
001225 "Subtitle (optional)", RT
001240 AT (15,54), FAC(HEX(81)), SUBTIT1$ , CH(20), RT
001255 AT (16,54), FAC(HEX(81)), SUBTIT2$ , CH(20), RT
001270 AT (18,25), RT
001285 "Starting word ", RT
001300 AT (18,54), FAC(HEX(81)), START$ , CH(15), RT
001315 AT (19,25), RT
001330 "Ending word ", RT
001345 AT (19,54), FAC(HEX(81)), END$ , CH(15), RT
001360 AT (20,25), RT
001375 "Lines per page", RT
001390 AT (20,54), LPP$ , PIC(##), RT
001405 AT (21,25), RT
001420 "Print at (TERMINAL/PRINTER)", RT
001435 AT (21,54), FAC(HEX(81)), TOP$ , CH(08), RT
001450 AT (23,30), RT
001465 "Press", RT
001480 AT (23,36), FAC(HEX(85)), ENTER$ , CH(05), RT
001495 AT (23,42), RT
001510 "to continue ", RT
001525 AT (24,15), RT
001540 "Press", RT
001555 AT (24,21), FAC(HEX(84)), PF16$ , CH(04), RT
001570 AT (24,26), RT
001585 "to return to the Glossary Maintenance Menu ", RT
001630 KEYS(BIN(0)&BIN(16)&BIN(32)),
001645 ON (BIN(16)&BIN(32)) GOTO EXIT,EXIT
001660
001675 ER$=" ":E$=HEX(8C):WELCOME$,HEADER$=" "
001690

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 15:18 11/13/84 Page 3  
Input File is CRESGLOP in Library CRES@SRC on Volume CRES02

```

001705 /* Before leaving this screen, be sure all values input are */ JM
001720 /* legal entries. */ JM
001722 IF POS(" X"=A$)>0 THEN 1735 RT
001724 ER$=" Please only use 'X' to specify the mandatory words." RT
001726 E$=HEX(94) RT
001727 GOTO PRISCREEN RT
001728 RT
001735 IF STR(TOP$,1,1) = "P" OR STR(TOP$,1,1) = "I" THEN LINESCK
001750 E$=HEX(94)
001765 ER$="Print at ' has only legal values TERMINAL or PRINTER."
001780 GOTO PRISCREEN
001795 LINESCK: IF LPP% < 51 THEN RANGECK
001810 ER$=" 'Lines per page' can be at most 50."
001825 E$=HEX(84)
001840 GOTO PRISCREEN
001855
001870 REX: ER$=" 'Starting word' comes after 'Ending word'."
001885 E$=HEX(94):GOTO PRISCREEN
001900 RANGECK: IF START$="FIRST" OR END$="LAST" THEN FORD JM
001915 IF START$>END$ THEN REX JM
001930
001945 /* Now it's time to generate our print file name */
001960
001975 FORD: DAT$= DATE JM
001990 D$= STR(DAT$,3,2)&"/"&STR(DAT$,5,2)&"/"&STR(DAT$,1,2)
002005 T$= STR(TIME,1,2)&":"&STR(TIME,3,2)
002020
002035 /* The use of this short screen is just to let the user know the */
002050 /* output file is being generated. */
002065 DISPLAY
002080 AT (01,03),
002095 "CRES: Glossary Print Subsystem", RT
002110 AT (01,61),
002125 "(Generation Screen)",
002140 AT (02,03),
002155 "=====
002170 =====",
002185 AT (12,23),
002200 "Output file now being generated!"
002215
002230 /* Create an new output file for this run */
002245 PFILE$="GLOSSY##" RT
002260 I$=0
002275 NEXT14: CONVERT I$ TO STR(PFILE$,1,2),PIC(00)
002290 I$=I$+1
002305 CALL "LIMITS" (PFILE$,SLIB$,SVOL$,CODE$,MSG$)
002320 IF CODE$="Y" THEN NEXT14
002335 OPEN NOGETPARM #2,OUTPUT,FILE=PFILE$,LIBRARY=SLIB$,
002350 VOLUME=SVOL$,SPACE=8000
002365
002425 GOSUB TIL PGE
002440 IF START$ <> "FIRST" THEN 2500 RT
002455 START$=" N" /* Recall the first value in */
002470 /* file is 'Mandatory Begin' */
002485 /* cal Glossary' */
002500 IF END$ = "LAST" THEN END$ = ALL(HEX(PF))
002515 PAGE%=0: RECS% = 1

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 15:18 11/13/84 Page 4  
Input File is CRESGLOP in Library CRES@SRC on Volume CRES02

```

002530      READ #1, KEY>=START$,IOBUFF$,EOD GOTO HOME          JM
002545      GOSUB HEADER
002560      IF STR(IOBUFF$,20)>END$ THEN HOME
002575 KNOT:
002576      IF A$=" " THEN 2590                                     RT
002577      IF A$="X" AND STR(IOBUFF$,210,1)="*" THEN 2590         RT
002579      GOTO AGAN                                               RT
002590      WRITE #2, USING IMG6,C$,STR(IOBUFF$,1,20),           :
002605      STR(IOBUFF$,210,1),STR(IOBUFF$,21,8),STR(IOBUFF$,45,55)
002620      RECS%=RECS%+1
002635      /* Only print the definition lines that are nonblank */
002650      IF STR(IOBUFF$,29,8)=" " AND STR(IOBUFF$,100,55)=" " THEN 2710 RT
002665      WRITE #2, USING IMG5,C$,STR(IOBUFF$,29,8),           :
002680      STR(IOBUFF$,100,55)
002695      RECS%=RECS%+1
002710      IF STR(IOBUFF$,37,8)=" " AND STR(IOBUFF$,155,55)=" " THEN 2755 RT
002725      WRITE #2, USING IMG5,C$,STR(IOBUFF$,37,8),           :
002740      STR(IOBUFF$,155,55)
002755      RECS%=RECS%+1
002770      IF RECS%>LPP% THEN RECS%=1
002785      IF RECS% = 1 THEN GOSUB HEADER
002800 AGAN:  READ #1, IOBUFF$,EOD GOTO HOME
002815      IF STR(IOBUFF$,20)>END$ THEN HOME
002830      GOTO KNOT
002845 HOME:  WRITE #2, USING FIT, C$
002860      WRITE #2, USING FIT, C$
002875      WRITE #2, USING IMG8, C$
002890      CLOSE#2
002905      IF STR(TOP$,1,1) ="P" THEN MESS /* If the printer was */
002920      CLOSE WS /* chosen for output */ JM
002935      MSG2$ = "CANCEL Processing" /* then notify user of*/
002950      MSK$=HEX(00) /* output file Id */
002965      CMSG$ = " " /* otherwise link to DISPLAY */
002980      CALL "LINK" ADDR("DISPFILE","P",RLIB$,RVOL$,3%,PFILE$,
002995      SLIB$,SVOL$, "C",MSG2$,27%, "H",MSK$,CMSG$,255%,C%,R%)
003010      ER$ = " " :E$=HEX(8C)
003025      IF C%<>0 OR R%<>0
003040      THEN ER$ = STR(CMSG$,141)
003055      GOTO PRINTSCRN
003070
003085 MESS: ER$="File "&PFILE$&" in library "&SLIB$&" on "&SVOL$&" :
003100      " generated." :E$=HEX(84)
003115      GOTO PRINTSCRN
003130
003145 /* Print the title page with subtitles supplied, starting words,*/ JM
003160 /* and ending words. */ JM
003175 TTL_PGE:
003190      WRITE #2, USING FIT, HEX(8001)
003205      WRITE #2, USING FIT, HEX(0015)
003220      C$ = HEX(0001)
003235      WRITE #2, USING FIT1, C$
003250      WRITE #2, USING FIT2, C$
003265      WRITE #2, USING FIT0, C$,D$,T$
003280      WRITE #2, USING FIT3, C$
003295      WRITE #2, USING FIT2, C$
003310      WRITE #2, USING FIT1, C$
003325      WRITE #2, USING FIT2, C$

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 15:18 11/13/84 Page 5  
Input File is CRESGLOP in Library CRES@SRC on Volume CRES02

```

003340      IF SUBTIT1$ = " " THEN GERRY
003355      WRITE #2, USING TIT4, C$, SUBTIT1$
003370      IF SUBTIT2$ <> " " THEN WRITE #2, USING :
003385 TIT10, C$, SUBTIT2$
003400      WRITE #2, USING TIT2, C$
003415 GERRY:
003430      WRITE #2, USING TIT8, C$, START$
003445      WRITE #2, USING TIT2, C$
003460      WRITE #2, USING TIT9, C$, ENDS$
003475      WRITE #2, USING TIT2, C$
003476      IF AS="X" THEN OPT$="MAND TECH WORDS ONLY" ELSE OPT$="ALL"
003480      WRITE #2, USING TIT5, C$, OPT$
003485      WRITE #2, USING TIT2, C$
003490      WRITE #2, USING TIT1, C$
003505      RETURN
003565 HEADER:
003580      PAGE#=PAGE#+1
003595      IF PAGE#=1 THEN FIRST
003610      WRITE #2, USING TIT, C$
003625 FIRST: WRITE #2, USING TIT, HEX(8001)
003640      IF SUBTIT1$ <> " " THEN FOUR
003655      WRITE #2, USING IMG, C$, D$, T$, PAGE#
003670      GO TO ALRITE
003685 FOUR: WRITE #2, USING IMG1, C$, SUBTIT1$, D$, T$, PAGE#
003700      IF SUBTIT2$ <> " " THEN WRITE #2, USING IMG7, C$, SUBTIT2$
003715 ALRITE: WRITE #2, USING IMG2, C$, LIB$, INVOL$
003730      WRITE #2, USING IMG3, C$
003745      WRITE #2, USING IMG4, C$
003760      WRITE #2, USING IMG5, C$
003775      WRITE #2, USING IMG3, C$
003790      RETURN
003805
003820 TIT0:### : ##### :
003835 ##### :
003850 TIT: ### :
003865 TIT1:### :-----:
003880 -----:
003895 TIT2:### : :
003910 :
003925 TIT3:### : ***** CRES GLOSSARY *!
003940 ***** :
003955 TIT4:### : TITLE OF LIST : #####! JM
003970 ##### :
003972 TIT5:### : OPTION : #####! JM
003974 ##### :
003985 TIT8:### : STARTING POINT : #####! JM
004000 ##### :
004015 TIT9:### : ENDING POINT : #####! JM
004030 ##### :
004045 TIT10:### : #####:
004060 ##### :
004075 IMG:### ##### :
004090 :
004105 PAGE ##### :
004105 IMG1:### TITLE OF LIST : ##### :
004120 ##### PAGE ##### :
004135 IMG7:### :
004150 IMG2:### Input File is GLOSSARY in Library ##### on Volume ###:

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 15:18 11/13/84 Page 6  
Input File is CRESGLOP in Library CRES@SRC on Volume CRES02

```

004165 ###
004180 IMG3:###=====
004195 =====
004210 IMG4:### Word          Mand Tech   File      Definition  RT
004225 IMG6:### #####          #####      #####      RT
004240 #####          #####          #####      RT
004255 IMG5:###          #####      #####      RT
004270 #####          #####          #####      RT
004285 IMG8:###   !!! END OF LISTING !!!
004300
004315 IOERR1: PRINT PAGE
004330 PRINT BELL, HEX(94); "ERROR in GLOSSARY list." JM
004345 FOR I=1 TO 500000 :NEXT I
004360 CLOSE #1
004375 GOTO PRISCREEN
004390 IOERR2: PRINT PAGE
004405 PRINT BELL, HEX(94); "ERROR in print file."
004420 FOR I=1 TO 500000 :NEXT I
004435 CLOSE #2
004450 GOTO PRISCREEN
004465 EXIT: CLOSE #1

```

## ROOT WORD EXPANSION SUBSYSTEM

### Root Word Expansion (CRES@ERW)

CRES@ERW in library CRES@SRC is the BASIC language program to perform the CRES root word expansions and merge the expanded forms into the CRES word lists. The object language version is CRES@ERW in library CRES@SAV.

In the CRES system, a root word is defined as the simplest form of a word, that form which cannot be reduced any further; i.e., the singular, present tense word for nouns, verbs and adjectives. Most other parts of speech have only one form, which is considered the root word. If the root word form of a word appears on one of the CRES word lists, it is usually desirable to have at least the more common expanded forms of that word included as well. This system will expedite these expansions. For the more common parts of speech, (nouns, verbs, and noun/verbs), the program automatically supplies suggested suffixes to form the plurals and past participles. The user has the option of eliminating any of the automatically formed expansions or correcting their spellings. In addition, up to 20 different forms of the root word may be generated by the user. After all expansions have been specified, the program will present them to the user, one at a time, for the part of speech and other attributes to be entered for each.

When the root word expansion is complete, the user can display the expansions on the terminal, print out a hard copy, and/or permanently merge the expansions into one of the CRES word lists. The program also allows the user to return and modify, delete from, or add to the expansions already specified. The list of expansions is contained in a work file, which may be retained on disk or deleted if no longer required.

Figure 3 presents a flowchart of the root word expansion program. A complete listing follows.

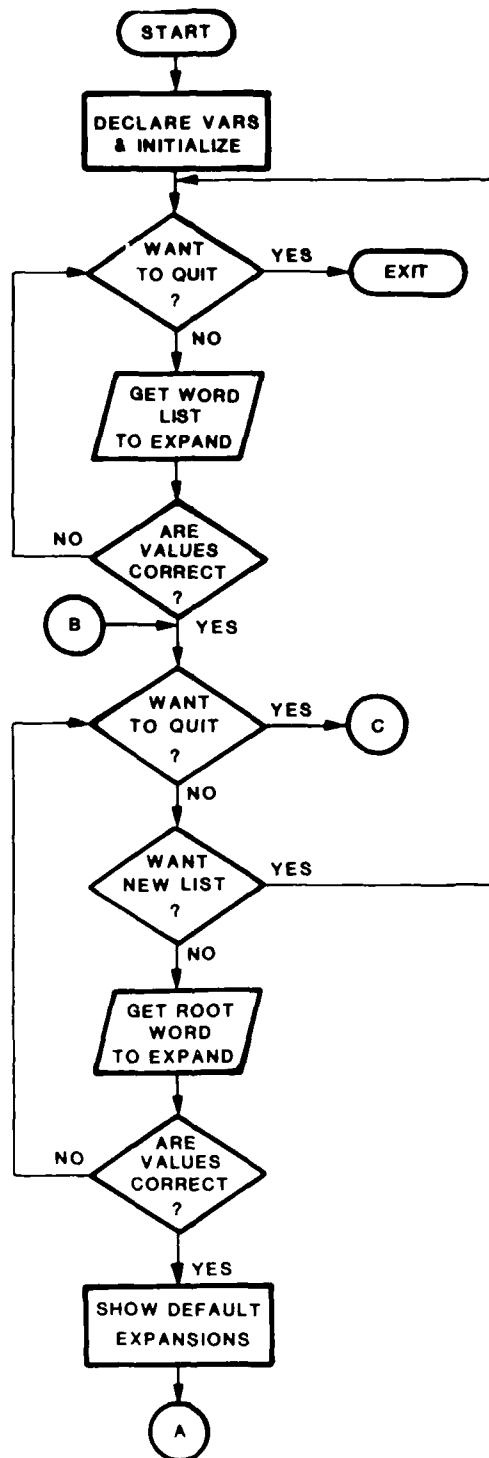


Figure 3. CRES Root Word Expansion Flowchart.

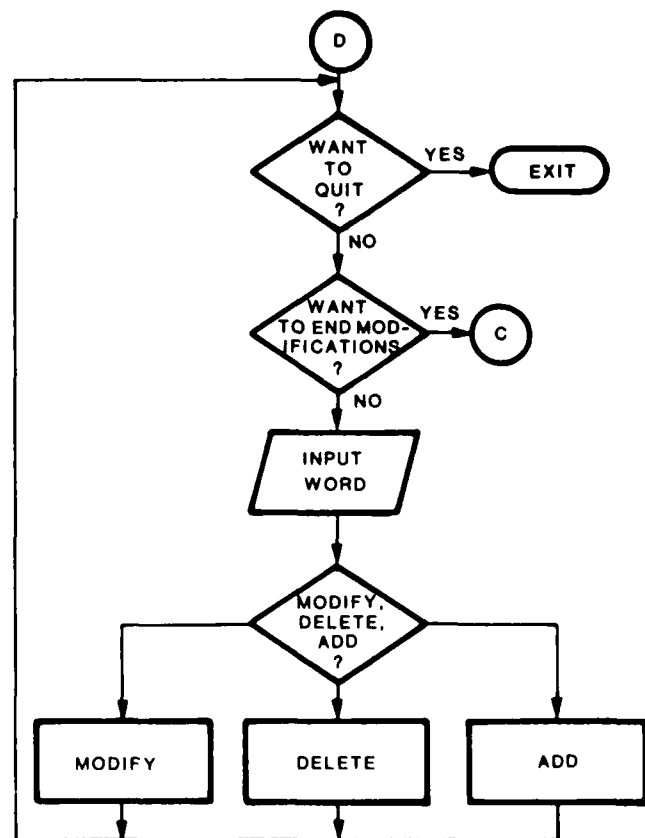
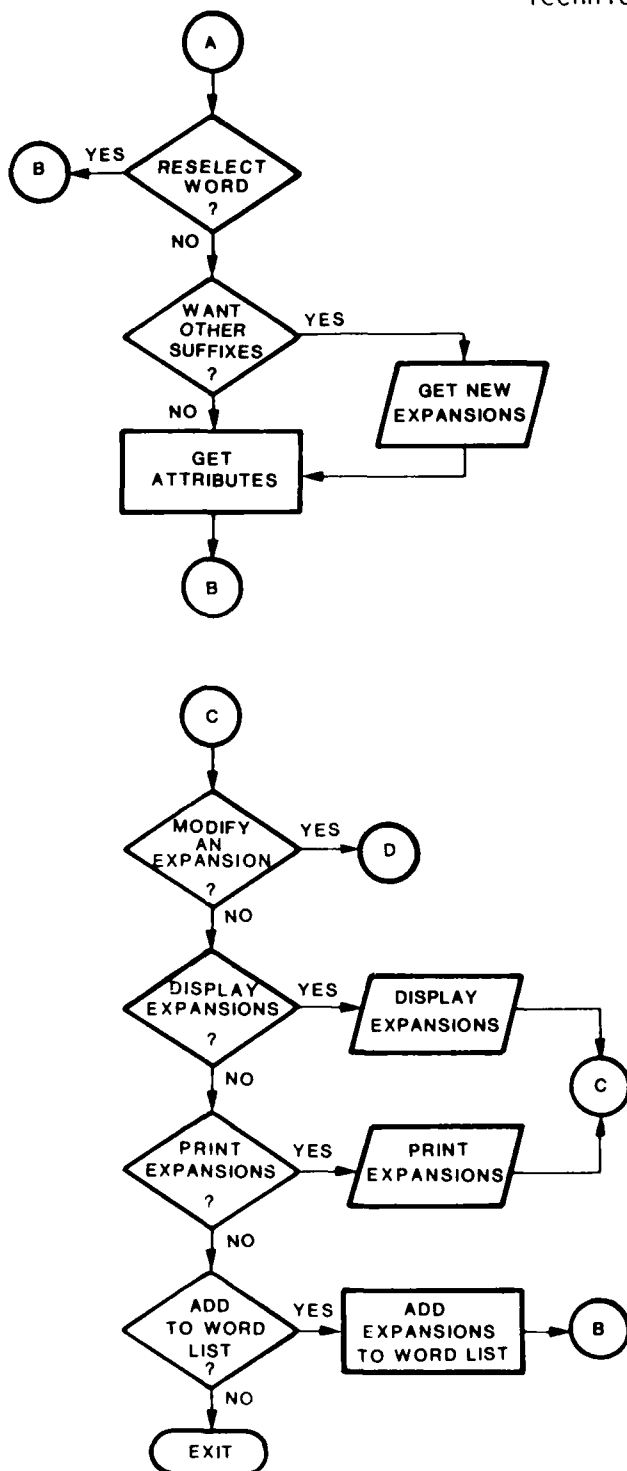


Figure 3 (Continued).

Technical Report 86-001

THIS PAGE INTENTIONALLY LEFT BLANK.

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 13:41 11/13/84 Page 1  
Input File is CRES@ERW in Library CRES@SRC on Volume CRES02

```

000010 /* */ RT
000015 /* PROGRAM CRES ROOT WORD EXPANSION SUBSYSTEM */ RT
000020 /* AUTHOR ROSE TANK */ RT
000025 /* DATE FEBRUARY 9, 1984 */ RT
000030 /* */ RT
000035 RT
000040 DIM FILE$8,LIB$8,VOL$6,DFIL$8,DLIB$8,DVOL$6,PFILE$8,BUFF$ RT
000045 DIM OPEN$(3),ROOTWD$20,PRIO$PC$1,EXPAND$(20)20,WORD$20,PPART$1 RT
000050 DIM NEG$1,TOB$1,ROOT$1,MIN$1,IOBUFF$40,AS(14)10,SLIB$8,SVOL$6 RT
000055 DIM HEADER$34,ITLE1$6,ITLE2$11,ENTER$5,PF1$3,PF2$3,PF15$4,PF16$4 JM
000060 DIM ITLE3$18,ITLE4$14,RTOEXP$10,MAND$24,ITLE5$14,ITLE6$10 JM
000065 DIM P$(13)3,EXP$9,PFKEY$6,PF$2,OPERATION$27,SUB1$20,SUB2$20,CODE$1 RT
000070 DIM ER$71,E$1,E$(20)1,ER1$65,E1$1,P$1,L$1,V$1,CMSG$255,MSK$2 RT
000071 DIM RLIB$8,RVOL$6,PASSWORD$9 JM
000075 RT
000080 F1$=HEX(A014A001A001A001A0010001A001A001000100030009) RT
000085 HEADER$="CRES ROOT WORD EXPANSION SUBSYSTEM" RT
000090 FILE$="CMDNDLST" RT
000091 CALL "EXTRACT" ADDR("IL",LIB$,"IV",VOL$,"SL",SLIB$,"SV",SVOL$, :
000093 "RL",RLIB$,"RV",RVOL$)
000105 ENTER$="ENTER" RT
000110 PF16$="PF16" RT
000115 ITLE1$="COMMON" JM
000120 ITLE2$="LESS COMMON" JM
000125 ENTER$="ENTER" RT
000130 PF1$="PF1" RT
000135 PF2$="PF2" RT
000140 PF15$="PF15" RT
000145 ITLE5$="PART OF SPEECH" JM
000150 ITLE6$="DEFINITION" JM
000155 P$(1)="{J}":P$(2)="{j}":P$(3)="{A}":P$(4)="{a}":P$(5)="{B}" RT
000160 P$(6)="{C}":P$(7)="{I}":P$(8)="{N}":P$(9)="{n}":P$(10)="{P}" RT
000165 P$(11)="{O}":P$(12)="{V}":P$(13)="{d}" RT
000170 EXP$="expansion" RT
000175 PFKEY$="PF-Key" RT
000180 PF$="PF" RT
000185 OPERATION$=" OPERATION" RT
000190 A1$="1":A2$="2":A3$="3":A15$="15":A16$="16" RT
000195 AS(1)="adjective" RT
000200 AS(2)="verb" RT
000205 AS(3)="conjunction" RT
000210 AS(4)="adj/verb" RT
000215 AS(5)="noun" RT
000220 AS(6)="pronoun" RT
000225 AS(7)="interject" RT
000230 AS(8)="noun/verb" RT
000235 AS(9)="adverb" RT
000240 AS(10)="preposit" RT
000245 AS(11)="article" RT
000250 AS(12)="adv/verb" RT
000255 AS(13)="noun/adj" RT
000260 RT
000265 SELECT #1, "WORDLIST", VARC, INDEXED, RECSIZE=40, KEYPOS=1, : RT
000270 KEYLEN=20, IOERR GOTO IOERR1 RT
000275 SELECT #2, "WDEXPAND", VARC, INDEXED, RECSIZE=40, KEYPOS=1, : RT
000280 KEYLEN=20, IOERR GOTO IOERR2 RT
000285 SELECT #3, "PRINTFIL", PRINTER, RECSIZE=80, IOERR GOTO IOERR3 RT

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 13:41 11/13/84 Page 2  
Input File is CRES@ERW in Library CRES@SRC on Volume CRES02

```

000290                                     RT
000295 SCREEN1: ER$=" ":F$,L$,V$ =HEX(81): E$=HEX(8C) RT
000300 FILMNU:                                     RT
000305 ACCEPT                                     : RT
000310         AT (01,03),                         : RT
000315 "CRES: Root Word Expansion Subsystem      (File Sele: RT
000320 ction Screen)",                             : RT
000325         AT (02,03),                         : RT
000330 "===== RT
000335 =====", RT
000340         AT (04,20),                         : RT
000345 "Welcome to", RT
000350         AT (04,31), FAC(HEX(85)), HEADER$, CH(34), : RT
000355         AT (06,03),                         : RT
000360 " This system allows the expansion of root words to be entered: RT
000365 into an", RT
000370         AT (07,03),                         : RT
000375 " existing word list with pre-defined or user-defined suffixes. : RT
000380 For the", RT
000385         AT (08,03),                         : RT
000390 " root words that are defined as the more common parts of speech : RT
000395 (noun,verbs,", RT
000400         AT (09,03),                         : RT
000405 " noun/verbs), this program will supply suggested suffixes. For : RT
000410 the other", RT
000415         AT (10,03),                        : RT
000420 " root words, you may enter up to 20 user-defined suffixes.", : RT
000425         AT (12,04),                        : RT
000430 "Please specify the word list in which you want to place the root: RT
000435 word", RT
000440         AT (13,04),                        : RT
000445 "expansions, if these defaults are not satisfactory.", : RT
000450         AT (16,33),                        : RT
000455 "File =", RT
000460         AT (16,43), FAC(F$), FILE$, CH(08), : RT
000465         AT (17,33),                        : RT
000470 "Library =", RT
000475         AT (17,43), FAC(L$), LIB$, CH(08), : RT
000480         AT (18,33),                        : RT
000485 "Volume =", RT
000490         AT (18,43), FAC(V$), VOL$, CH(06), : RT
000495         AT (20,19), FAC(E$), ER$, CH(46), : RT
000500         AT (23,11),                        : RT
000505 "Press", RT
000510         AT (23,17), FAC(HEX(85)), ENTER$, CH(05), : RT
000515         AT (23,23),                        : RT
000520 "to continue or", RT
000525         AT (23,38), FAC(HEX(84)), PF16$, CH(04), : RT
000530         AT (23,43),                        : RT
000535 "to return to the Master Menu", RT
000540 KEYS(BIN(0)&BIN(16)&BIN(32)), RT
000545 ON (BIN(16)&BIN(32)) GOTO EXIT,EXIT RT
000550 RT
000555 ER$=" ": E$=HEX(8C) RT
000560 /* SEE IF THE SPECIFIED FILE EXISTS */ RT
000565 /* Y- the file exists and the program may continue */ JM
000570 /* N- the file doesn't currently exist; you may create one */ JM

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 13:41 11/13/84 Page 3  
Input File is CRES0ERW in Library CRES0SRC on Volume CRES02

```

000575 /* V- the volume supplied doesn't exist or is not mounted */ RT
000580 /* L- the library supplied doesn't exist in the volume given */ RT
000585 /* E- there is an I/O error in the file specified */ RT
000590 RT
000595 ER$="File not found in Library/Volume." RT
000600 CALL "LIMITS"(FILE$,LIB$,VOL$,CODE$,ER$) RT
000605 IF CODE$="Y" THEN RECC RT
000610 IF CODE$="N" THEN GOSUB'7(ER$,HEX(91),HEX(81),HEX(81)) RT
000615 IF CODE$="V" THEN GOSUB'7(STR(ER$,3),HEX(81),HEX(81),HEX(91)) RT
000620 IF CODE$="L" THEN GOSUB'7(STR(ER$,3),HEX(81),HEX(91),HEX(81)) RT
000625 IF CODE$="E" THEN GOSUB'7(STR(ER$,3),HEX(81),HEX(81),HEX(81)) RT
000630 GOTO FILMNU RT
000635 IOERR1: PRINT PAGE RT
000640 PRINT HEX(94),"Error occurred on file "; FILE$ RT
000645 PRINT HEX(94),"Error code = "; FS(#1) RT
000650 STOP "Touch ENTER to return to menu.":CLOSE #1 RT
000655 OPEN#(1)=0: GOTO FILMNU RT
000660 RT
000665 IOERR2: PRINT PAGE RT
000670 PRINT HEX(94),"Error occurred on file "; FILE$ RT
000675 PRINT HEX(94),"Error code = "; FS(#2) RT
000680 STOP "Touch ENTER to return to menu.":CLOSE #2 RT
000685 OPEN#(2)=0: GOTO FILMNU RT
000690 RT
000695 IOERR3: PRINT PAGE RT
000700 PRINT HEX(94),"Error occurred on print file "; FILE$ RT
000705 PRINT HEX(94),"Error code = "; FS(#3) RT
000710 STOP "Touch ENTER to return to menu.":CLOSE #3 RT
000715 OPEN#(3)=0: GOTO FILMNU RT
000720 RT
000725 RT
000730 RECC: CALL "READFOR" ADDR(FILE$,LIB$,VOL$,0$, "RS",RSIZE$,RET$) RT
000735 IF RSIZE$=40 THEN OK RT
000740 ES=HEX(94) RT
000745 ER$= "File type is wrong - please respecify." RT
000750 GOTO FILMNU RT
000755 RT
000760 OK:/* Create a new data file for expansions. */ RT
000765 IF FLAG=1 THEN 835 RT
000770 DFIL$="ROOT##": DLIB$=LIB$ :DVOL$=VOL$ :I$=0 RT
000775 C: CONVERT I$ TO STR(DFIL$,5,2), PIC(00) RT
000780 CALL "LIMITS"(DFIL$,DLIB$,DVOL$,RC$) RT
000785 I$=I$+1 RT
000790 IF RC$="Y" THEN C RT
000795 IF OPEN#(2)=1 THEN CLOSE #2 RT
000800 OPEN NOGETPARM #2, OUTPUT, FILE=DFIL$, LIBRARY=DLIB$, : RT
000805 VOLUME=DVOL$,SPACE=100 RT
000810 CLOSE #2 RT
000815 OPEN NOGETPARM #2, IO, SPACE=100, FILE=DFIL$, LIBRARY=DLIB$, : RT
000820 VOLUME=DVOL$ RT
000825 OPEN#(2)=1 RT
000830 RT
000835 IF OPEN#(1)=1 THEN CLOSE #1 RT
000840 OPEN NOGETPARM #1,SHARED, FILE= FILE$, LIBRARY = LIB$, : RT
000845 VOLUME = VOL$ RT
000850 OPEN#(1)=1 RT
000855 RT

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 13:41 11/13/84 Page 4  
Input File is CRES@ERW in Library CRES@SRC on Volume CRES02

```

000860 READ #1,IOBUFF$ RT
000865 IF IOBUFF$=" COMMON WORDS" THEN 875 RT
000870 MAND$="Mandatory Technical Word": M$=HEX(81):GOTO 885 RT
000875 MAND$=" ": M$=HEX(8D) RT
000880 RT
000885 IOBUFF$=" " :E1$=HEX(8C):ER1$=" " RT
000890 SCREEN2: ER$,PR1OSPC$,ROOTWD$=" " : EXPAND$()=" " : E$=HEX(81) RT
000895 E$=HEX(8C) RT
000897 PASSWORD$=HEX(00010F10111F20) JM
000900 WDSLEFT: WDNO%=0 JM
000905 ACCEPT : RT
000910 AT (01,03), RT
000915 "CRES: Root Word Expansion Subsystem (Word Specific) RT
000920 ation Screen)", RT
000925 AT (02,03), RT
000930 "===== RT
000935 =====", RT
000940 AT (04,02), JM
000945 " Please enter a root word and select, by letter, its correspondi RT
000950 ng part of", RT
000955 AT (05,03), RT
000960 "speech as listed in either the common or less common category.", RT
000965 AT (07,26), RT
000970 "Root word =", RT
000975 AT (07,39), ROOTWD$ , CH(20), RT
000980 AT (08,26), RT
000985 "Part of Speech", RT
000990 AT (08,41),FAC(HEX(80)), PR1OSPC$ , CH(01), RT
000995 AT (09,08), FAC(E$), ER$ , CH(65), JM
001000 AT (10,21),FAC(HEX(AD)), IFLE1$ , CH(06), JM
001005 AT (10,55),FAC(HEX(AD)), IFLE2$ , CH(11), JM
001010 AT (11,18), JM
001015 "N =Noun B =Article", RT
001020 AT (12,18), JM
001025 "V =Verb I =Interjection", RT
001030 AT (13,18), JM
001035 "n =Noun/verb J =Adjective", RT
001040 AT (14,52), JM
001045 "C =Conjunction", RT
001050 AT (15,52), JM
001055 "P =Preposition", RT
001060 AT (16,52), JM
001065 "a =Adverb/verb", RT
001070 AT (17,52), JM
001075 "A =Adverb", RT
001080 AT (18,52), JM
001085 "j =Adjective/verb", RT
001090 AT (19,52), JM
001095 "d =Noun/adjective", RT
001100 AT (21,30), RT
001105 "Press", RT
001110 AT (21,36), FAC(HEX(85)), ENTER$ , CH(05), RT
001115 AT (21,42), RT
001120 "to continue", RT
001125 AT (22,13), RT
001130 "Press", RT
001135 AT (22,19), FAC(HEX(84)), PF1$ , CH(03), RT

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 13:41 11/13/84 Page 5  
Input File is CRES0ERW in Library CRES0SRC on Volume CRES02

```

001140      AT (22,23),                      : RT
001145 "to view the definitions of the parts of speech", : RT
001150      AT (23,17),                      : RT
001155 "Press",                             : RT
001160      AT (23,23),FAC(HEX(84)),          PF15$      , CH(04), : RT
001165      AT (23,28),                      : RT
001170 "to return to the List Selection Screen", : RT
001175      AT (24,28),                      : RT
001180 "Press",                             : RT
001185      AT (24,34),FAC(HEX(84)),          PF16$      , CH(04), : RT
001190      AT (24,39),                      : RT
001195 "to end expansions",                 : RT
001200 KEYS(PASSWORD$),                     : JM
001205      KEY(OPTION),                     : JM
001210 ON(BIN(15)&BIN(31)&BIN(16)&BIN(32)) GOTO BLANKF,BLANKF, : RT
001215      PRINT SCREEN_INIT,PRINT_SCREEN_INIT : RT
001220 IF OPTION<>1 AND OPTION<>17 THEN 1235 : RT
001225     GOSUB DEFN: GOTO WDSLT           : RT
001230                                     : RT
001235 ER$=" ":E$=HEX(8C):E$=HEX(81):WDNO%=0 : RT
001240 IF OPTION=2 OR OPTION=18 THEN UNCOMMON : RT
001245 FOR I= 1 TO LEN(ROOTWD$)           : RT
001250     IF POS("ABCDEFGHIJKLMNORSTUVWXYZ'--"=STR(ROOTWD$,I,1))>0 : JM
001255     THEN 1270                         : RT
001260     ER$="Illegal character was found in the root word-please reenter" : RT
001265     E$=HEX(94): GOTO WDSLT         : RT
001270 NEXT I                             : RT
001275                                     : RT
001280 K= POS("NVnABCDEFGHIJpajd"=PRTO$PC$) : RT
001285 IF K<>0 THEN 1300                   : RT
001290 ER$="Part of speech is invalid - please respecify." : JM
001295 E$=HEX(94): GOTO WDSLT             : RT
001300 READ #2, KEY=ROOTWD$, IOBUFF$,EOD GOTO NOT FOUND : RT
001305 ER$= ROOTWD$&" is already in the list - press PF2 for more expans! : JM
001310 tons.":E$=HEX(94)                  : JM
001312 PASSWORD$=HEX(0001020F1011121F20) : JM
001313 GOTO WDSLT                         : JM
001315 NOT_FOUND: IF K<4 THEN EXPANSIONS ELSE UNCOMMON : RT
001320                                     : RT
001325 BLANKF: FILE$=" ": CLOSE #1: OPEN%(1)=0 : RT
001330     IF OPEN%(2)=0 THEN SCREEN1       : RT
001331     CLOSE #2                          : RT
001332     CALL "SCRATCH" ADDR("F",DFIL$,DLIB$,DVOL$,"a") : RT
001333     OPEN%(2)=0                        : RT
001334     GOTO SCREEN1                     : RT
001335                                     : RT
001340 EXPANSIONS:                          : RT
001345 EXPAND$()=" "                       : RT
001350 RT_LEN=LEN(ROOTWD$)                 : RT
001355 IF STR(ROOTWD$,RT_LEN)<>"Y" THEN 1390 : RT
001360     EXPAND$(1)=STR(ROOTWD$,,RT_LEN-1)&"IES" : RT
001365     IF K= 1 THEN 1450               : RT
001370     EXPAND$(2)=STR(ROOTWD$,,RT_LEN-1)&"IED" : RT
001375     EXPAND$(3)=ROOTWD$&"ING"       : RT
001380     GOTO 1450                        : RT
001385                                     : RT
001390 IF STR(ROOTWD$,RT_LEN)<>"E" THEN 1425 : RT

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 13:41 11/13/84 Page 6  
Input File is CRES@ERW in Library CRES@SRC on Volume CRES02

```

001395 EXPAND$(1)=" " RT
001400 IF K= 1 THEN 1450 RT
001405 EXPAND$(2)=STR(ROOTWD$,RT_LEN)&"D" RT
001410 EXPAND$(3)=STR(ROOTWD$,RT_LEN-1)&"ING" RT
001415 GOTO 1450 RT
001420 RT
001425 EXPAND$(1)=ROOTWD$&"ES" RT
001430 IF K= 1 THEN 1450 RT
001435 EXPAND$(2)=ROOTWD$&"ED" RT
001440 EXPAND$(3)=ROOTWD$&"ING" RT
001445 RT
001450 EXPAND$(4)=ROOTWD$&"S" RT
001455 RT
001460 E$( )=HEX(81) RT
001465 COMMON: RT
001470 ACCEPT : RT
001475 AT (01,03), : RT
001480 "CRES: Root Word Expansion Subsystem (Common Category Expans RT
001485 nsion Screen)", : RT
001490 AT (02,03), : RT
001495 "===== RT
001500 =====", : RT
001505 AT (04,04), : RT
001510 "These are the default expansions for the common root word you've RT
001515 entered.", : RT
001520 AT (05,04), : RT
001525 "Please modify or delete any incorrect expansions.", : RT
001530 AT (08,21), : RT
001535 "The root word was:", : RT
001540 AT (08,42), FAC(HEX(85)) , ROOTWD$ , CH(20), : RT
001545 AT (09,21), : RT
001550 "Part of speech was:", : RT
001555 AT (09,42), FAC(HEX(84)) , PRTOSPC$ , CH(1), : RT
001560 AT (13,23), : RT
001565 "-S expansion is", : RT
001570 AT (13,41), FAC(E$(4)) , EXPAND$(4) , CH(20), : RT
001575 AT (14,23), : RT
001580 "-ES expansion is", : RT
001585 AT (14,41), FAC(E$(1)) , EXPAND$(1) , CH(20), : RT
001590 AT (15,23), : RT
001595 "-ED expansion is", : RT
001600 AT (15,41), FAC(E$(2)) , EXPAND$(2) , CH(20), : RT
001605 AT (16,23), : RT
001610 "-ING expansion is", : RT
001615 AT (16,41), FAC(E$(3)) , EXPAND$(3) , CH(20), : RT
001620 AT (19,12), FAC(E$) , ER$ , CH(55), : RT
001625 AT (22,30), : RT
001630 "Press", : RT
001635 AT (22,36), FAC(HEX(85)), ENTER$ , CH(05), : RT
001640 AT (22,42), : RT
001645 "to continue", : RT
001650 AT (23,23), : RT
001655 "Press", : RT
001660 AT (23,29), FAC(HEX(84)), PF2$ , CH(03), : RT
001665 AT (23,33), : RT
001670 "to add user-defined suffixes", : RT
001675 AT (24,15), : RT

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 13:41 11/13/84 Page 7  
Input File is CRES0ERW in Library CRES0SRC on Volume CRES02

```

001680 "Press",                                     : RT
001685 AT (24,21), PAC(HEX(84)), PF15$ , CH(04), : RT
001690 AT (24,26),                                     : RT
001695 "to reselect the word and/or part of speech", : RT
001700 KEYS(BIN(0)&BIN(2)&BIN(18)&BIN(15)&BIN(31)), KEY(OPTION), : RT
001705 ON(BIN(15)&BIN(31)) GOTO SCREEN2,SCREEN2      : RT
001710                                     : RT
001715 ER$=" ": ES=HEX(8C): ES(1)=HEX(81)          : RT
001720 FOR I= 1 TO 4                                     : RT
001725 IF EXPAND$(I)=" " THEN 1760                  : RT
001730 FOR J= 1 TO LEN(EXPAND$(I))                   : RT
001735 IF POS,"ABCDEFGHIJKLMNPOQRSTUVWXYZ'-'=STR(EXPAND$(I),J,1))>0 : JM
001740 THEN 1755                                         : RT
001745 ER$="Illegal character found in expansion-please modify." : RT
001750 ES=HEX(84): ES(I)=HEX(91): GOTO COMMON        : RT
001755 NEXT J                                           : RT
001760 NEXT I : GOTO 1775                             : RT
001770                                     : RT
001775 WDN0%=4                                           : RT
001780 IF OPTION=2 OR OPTION=18 THEN UNCOMMON        : RT
001785                                     : RT
001790 ADD EXP:                                         : RT
001795 IF ROOTWD$=" " THEN 1875                       : RT
001800 READ #2,KEY=ROOTWD$,EOD GOTO 1820             : RT
001805 ER$=ROOTWD$&" has already been added to the expansion list." : RT
001810 ES=HEX(94): GOTO 1875                           : RT
001815                                     : RT
001820 RTOEXP$="root word:" : WORD$= ROOTWD$: ROOT$="**" : RT
001825 PPART$,NEGS$,TOBES$,MIW$=" "                  : RT
001830 GOSUB ATTRIBUTES                                : RT
001835 IF OPTION=0 THEN 1855                          : RT
001836 ER$=WORD$ &" has been deleted from the expansions." : JM
001840 *ER$=EXPAND$(I)&" has been deleted from the expansions." : JM
001845 GOTO 1875                                         : RT
001850                                     : RT
001855 $PACK (P=PF15) IOBUFF$ FROM WORD$, PRTO$PC$, PPART$, : RT
001860 NEGS$,TOBES$,ROOT$,MIW$                          : RT
001865 WRITE #2, IOBUFF$, DATA GOTO IOERR2           : RT
001870                                     : RT
001875 RTOEXP$="Expansion:"                             : RT
001880                                     : RT
001885 FOR I= 1 TO WDN0%                               : RT
001890 IF EXPAND$(I)= " " THEN 1980                   : RT
001895                                     : RT
001900 READ #2,KEY=EXPAND$(I),EOD GOTO 1920          : RT
001905 ER$=EXPAND$(I)&" has already been added to the expansion list." : RT
001910 ES=HEX(8C): GOTO 1980                             : RT
001915                                     : RT
001920 PRTO$PC$,NEGS$,TOBES$,ROOT$,MIW$=" "          : RT
001925 L=LEN(EXPAND$(I))                                : RT
001930 IF STR(EXPAND$(I),L-1,L)="ED" THEN 1932        : RT
001931 PPART$=" ": GOTO 1935                             : RT
001932 PPART$="**": PRTO$PC$="V"                       : RT
001935 WORD$=EXPAND$(I): GOSUB ATTRIBUTES                : RT
001940 IF OPTION=2 OR OPTION=18 THEN 1975             : RT
001945                                     : RT
001950 $PACK (P=PF15) IOBUFF$ FROM WORD$, PRTO$PC$, PPART$, : RT

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 13:41 11/13/84 Page 8  
Input File is CRES@ERW in Library CRES@SRC on Volume CRES02

```

001955      NEG$,TOBE$,ROOT$,MTW$      RT
001960      WRITE #2, IOBUFF$, DATA GOTO IOERR2      RT
001965      GOTO 1980      RT
001970      RT
001975      ER$=EXPAND$(I)&" has been deleted from the expansions."      RT
001980      NEXT I      RT
001985      RT
001990      GOTO SCREEN2      RT
001995      RT
002000      UNCOMMON:      RT
002005      ACCEPT      RT
002010      AT (01,03),      RT
002015      "CRES: Root Word Expansion Suosystem      (User-defined S: RT
002020      uffix Screen)",      RT
002025      AT (02,03),      RT
002030      "=====      RT
002035      =====",      RT
002040      AT (04,04),      RT
002045      " The fields below have been provided for user-defined suffixes." RT
002050      Please",      RT
002055      AT (05,04),      RT
002060      "enter any",      RT
002065      AT (05,14), FAC(HEX(84)),      EXP$      , CH(09),      RT
002070      AT (05,24),      RT
002075      "you want to appear in",      JM
002080      AT (05,46), FAC(HEX(85)),      FILE$      , CH(08),      RT
002085      AT (05,55),      RT
002090      ". This screen has been",      RT
002095      AT (06,04),      RT
002100      "provided to enable you to enter the less common expansions of yo: RT
002105      ur choice.",      RT
002110      AT (10,26),      RT
002115      "Expansion 1",      RT
002120      AT (10,38), FAC(E$(WDNO%+1 )), EXPAND$(WDNO%+1)      , CH(20),      RT
002125      AT (11,26),      RT
002130      "Expansion 2",      RT
002135      AT (11,38), FAC(E$( WDNO%+2 )), EXPAND$(WDNO%+2)      , CH(20),      RT
002140      AT (12,26),      RT
002145      "Expansion 3",      RT
002150      AT (12,38), FAC(E$( WDNO%+3 )), EXPAND$(WDNO%+3)      , CH(20),      RT
002155      AT (13,26),      RT
002160      "Expansion 4",      RT
002165      AT (13,38), FAC(E$( WDNO%+4 )), EXPAND$(WDNO%+4)      , CH(20),      RT
002170      AT (14,26),      RT
002175      "Expansion 5",      RT
002180      AT (14,38), FAC(E$( WDNO%+5 )), EXPAND$(WDNO%+5)      , CH(20),      RT
002185      AT (18,14), FAC(E$),      ER$      , CH(56),      RT
002190      AT (22,30),      RT
002195      "Press",      RT
002200      AT (22,36), FAC(HEX(85)),      ENTER$      , CH(05),      RT
002205      AT (22,42),      RT
002210      "to continue",      RT
002215      AT (23,20),      RT
002220      "Press",      RT
002225      AT (23,26), FAC(HEX(84)),      PF2$      , CH(03),      RT
002230      AT (23,30),      RT
002235      "to add more user-defined suffixes",      RT

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 13:41 11/13/84 Page 9  
Input File is CRES0ERW in Library CRES0SRC on Volume CRES02

```

002240      AT (24,15),                      : RT
002245 "Press",                             : RT
002250      AT (24,21), FAC(HEX(84)),          PF15$      , CH(04), : RT
002255      AT (24,26),                      : RT
002260 "to reselect the word and/or part of speech" , : RT
002265 KEYS(BIN(0)&BIN(2)&BIN(18)&BIN(15)&BIN(31)), KEY(OPTION), : RT
002270 ON (BIN(15)&BIN(31)) GOTO SCREEN2,SCREEN2 : RT
002275                                         : RT
002280                                         : RT
002285 ER$=" ":ES=HEX(8C):E$(I)=HEX(81)      : RT
002290 FOR I=(WDNO%+1) TO (WDNO%+5)           : RT
002295 IF EXPAND$(I)=" " THEN 2330            : RT
002300 FOR J= 1 TO LEN(EXPAND$(I))           : RT
002305 IF POS("ABCDEFGHIJKLMNOPSUVWXYZ'-'=STR(EXPAND$(I),J,1))>0 : JM
002310 THEN 2325                               : RT
002315 ER$="Illegal character found in expansion - please modify." : RT
002320 ES=HEX(84): E$(I)=HEX(91): GOTO UNCOMMON : RT
002325 NEXT J                                 : RT
002330 NEXT I                                 : RT
002335                                         : RT
002340 IF WDNO%<11 THEN WDNO%=WDNO%+5        : JM
002341 IF I>=20 AND(OPTION=2 OR OPTION=18) THEN ER$="The system allows o : JM
002342 nly 20 expansions per word per trial." : JM
002345 IF OPTION=2 OR OPTION=18 THEN UNCOMMON : RT
002350 GOTO ADD_EXP                           : RT
002360                                         : RT
002365 ATTRIBUTES:                           : RT
002370 IF MAND$<>" THEN ITLE9$="(* or NO ENTRY)" ELSE ITLE9$=" " : JM
002375 ITLE3$      ="CURRENT ATTRIBUTES"      : JM
002380 ITLE4$      ="LEGAL ENTRIES"           : JM
002385 ACCEPT                                         : RT
002390      AT (01,03),                          : RT
002395 "CRES: Root Word Expansion Subsystem      (Attribute O : RT
002400 ption Screen)",                          : RT
002405      AT (02,03),                          : RT
002410 "===== : RT
002415 =====",                          : RT
002420      AT (04,03),                          : RT
002425 "Please specify the proper attributes for this expansion to be en : RT
002430 tered into",                          : RT
002435      AT (05,03), FAC(HEX(85)),          FILE$      , CH(08), : RT
002440      AT (05,12),                          : RT
002445 ". ",                          : RT
002450      AT (07,30), FAC(HEX(AD)),          ITLE3$      , CH(18), : JM
002455      AT (07,60), FAC(HEX(AD)),          ITLE4$      , CH(13), : JM
002460      AT (09,07), FAC(HEX(8C)),          RTOEXP$     , CH(10), : RT
002465      AT (09,20), FAC(HEX(85)),          WORD$       , CH(20), : RT
002470      AT (09,60),                          : RT
002475 " (A-Z-')",                          : JM
002480      AT (11,07),                          : RT
002485 "Part of Speech",                      : RT
002490      AT (11,39), FAC(HEX(80)),          PRTOSPC$    , CH(01), : RT
002495      AT (11,60),                          : RT
002500 " (ABCDEFGHIJNOPVajnd)",                : RT
002505      AT (12,07),                          : RT
002510 "Past Participle",                    : RT
002515      AT (12,39), PPART$      , CH(01), : RT

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 13:41 11/13/84 Page 10  
Input File is CRES00ERW in Library CRES0SRC on Volume CRES02

```

002520      AT (12,60),      : RT
002525 "(* or NO ENTRY)",  : RT
002530      AT (13,07),      : RT
002535 "Negative",          : RT
002540      AT (13,39), NEG$      , CH(01), : RT
002545      AT (13,60),          : RT
002550 "(* or NO ENTRY)",  : RT
002555      AT (14,07),          : RT
002560 "To Be",              : RT
002565      AT (14,39), TOBE$     , CH(01), : RT
002570      AT (14,60),          : RT
002575 "(* or NO ENTRY)",  : RT
002580      AT (15,07),          : RT
002585 "Root Word",        : RT
002590      AT (15,39), ROOT$      , CH(01), : RT
002595      AT (15,60),          : RT
002600 "(* or NO ENTRY)",  : RT
002605      AT (16,07), FAC(HEX(8C)), MAND$      , CH(24), : RT
002610      AT (16,39), FAC(M$),      MIW$      , CH(01), : RT
002615      AT (16,60),          : RT
002620      FAC(HEX(8C)), ITLE9$,CH(15), : JM
002625      AT (18,10), FAC(E$),      ER$      , CH(60), : RT
002630      AT (21,30),          : RT
002635 "Press",            : RT
002640      AT (21,36), FAC(HEX(85)), ENTER$      , CH(05), : RT
002645      AT (21,42),          : RT
002650 "to continue",      : RT
002655      AT (22,13),          : RT
002660 "Press",            : RT
002665      AT (22,19), FAC(HEX(84)), PF1$      , CH(03), : RT
002670      AT (22,23),          : RT
002675 "to view the definitions of the parts of speech", : RT
002680      AT (23,24),          : RT
002685 "Press",            : RT
002690      AT (23,30), FAC(HEX(84)), PF2$      , CH(04), : RT
002695      AT (23,34),          : JM
002700 "to delete this expansion", : RT
002705      AT (24,22),          : RT
002710 "Press",            : RT
002715      AT (24,28), FAC(HEX(84)), PF16$     , CH(04), : RT
002720      AT (24,33),          : RT
002725 "to return to the Master Menu", : RT
002730 KEYS(BIN(0)&BIN(1)&BIN(17)&BIN(2)&BIN(18)&BIN(16)&BIN(32)), : RT
002735 KEY (OPTION),          : RT
002740 ON(BIN(16)&BIN(32)) GOTO SCRATCH_FILE,SCRATCH_FILE : RT
002745 IF OPTION =1 OR OPTION =17 THEN GOSUB DEFN : RT
002750 IF OPTION =1 OR OPTION =17 THEN GOTO ATTRIBUTES : RT
002755 : RT
002760 IF OPTION=2 OR OPTION=18 THEN RETURN : RT
002765 ER$=" ":E$=HEX(8C) : RT
002770 /* Check for validity of entries */ : RT
002775 : RT
002780      IF POS("ABCDEFGHIJNOPVadjn"=PR10SPC$)>0 THEN NEXT1 : RT
002785      ER$ = "Part of Speech value is illegal." : RT
002790      E$ = HEX(94) : RT
002795      GOTO ATTRIBUTES : RT
002800 : RT

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 13:41 11/13/84 Page 11  
Input File is CRES00RW in Library CRES0SRC on Volume CRES02

```

002805 NEXT1: IF POS(" **=PPART$)>0 THEN NEXT2 RT
002810 ER$ = "Past Participle value is illegal." RT
002815 E$ = HEX(94) RT
002820 GOTO ATTRIBUTES RT
002825 NEXT2: IF POS(" **=NEG$)>0 THEN NEXT3 RT
002830 ER$ = " Negative value is illegal." RT
002835 E$ = HEX(94) RT
002840 GOTO ATTRIBUTES RT
002845 NEXT3: IF POS(" **=TOBE$)>0 THEN NEXT4 RT
002850 ER$ = " To Be value is illegal." RT
002855 E$ = HEX(94) RT
002860 GOTO ATTRIBUTES RT
002865 NEXT4: IF POS(" **=MTW$)>0 THEN NEXT5 RT
002870 ER$ = "Mandatory word value is illegal." RT
002875 E$ = HEX(94) RT
002880 GOTO ATTRIBUTES RT
002885 NEXT5: IF POS(" **=ROOT$)>0 THEN 2905 RT
002890 ER$ = " Root word value is illegal." RT
002895 E$ = HEX(94) RT
002900 GOTO ATTRIBUTES RT
002905 RETURN RT
002910 RT
002915 DEFN: RT
002920 ACCEPT
002925 AT (01,01),
002930 "CRES: Root word Expansion Subsystem (Definition of JM
002935 Entries Screen)", JM
002940 AT (02,01), "=====
002945 =====",
002950 AT (05,08), FAC(HEX(AD)), IFLE5$ , CH(14), JM
002955 AT (05,49), FAC(HEX(AD)), IFLE6$ , CH(10), JM
002960 AT (07,03), FAC(HEX(84)), P$(1) , CH(03),
002965 AT (07,08),
002970 "Adjective .....Modifies a noun or pronoun",
002975 AT (08,03), FAC(HEX(84)), P$(2) , CH(03),
002980 AT (08,08),
002985 "Adjective/verb .....A word that may function as an adjective
002990 or verb",
002995 AT (09,03), FAC(HEX(84)), P$(3) , CH(03),
003000 AT (09,08),
003005 "Adverb .....Modifies a verb, adjective, or adverb",
003010 AT (10,03), FAC(HEX(84)), P$(4) , CH(03),
003015 AT (10,08),
003020 "Adverb/verb .....A word that may function as an adverb or
003025 verb",
003030 AT (11,03), FAC(HEX(84)), P$(5) , CH(03),
003035 AT (11,08),
003040 "Article .....A limiting adjective before a noun (e.g.
003045 a, the)",
003050 AT (12,03), FAC(HEX(84)), P$(6) , CH(03),
003055 AT (12,08),
003060 "Conjunction .....Links phrases or clauses (e.g. or)",
003065 AT (13,03), FAC(HEX(84)), P$(7) , CH(03),
003070 AT (13,08),
003075 "Interjection .....Used for simple exclamations (e.g. Oh!)",
003080 ,
003085 AT (14,03), FAC(HEX(84)), P$(8) , CH(03),

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 13:41 11/13/84 Page 12  
Input File is CRES@ERW in Library CRES@SRC on Volume CRES02

```

003090      AT (14,08),
003095 "Noun .....Indicates a person, place, or thing",
003100      AT (15,03),  FAC(HEX(84)),    P$(13)      , CH(03),
003105      AT (15,08),
003110 "Noun/adjective .....A word that may function as a noun or ad-
003115 jective",
003120      AT (16,03),  FAC(HEX(84)),    P$(9)        , CH(03),
003125      AT (16,08),
003130 "Noun/verb .....Word that may function as a noun or verb",
003135 ",
003140      AT (17,03),  FAC(HEX(84)),    P$(10)       , CH(03),
003145      AT (17,08),
003150 "Preposition .....Expresses a relationship between two ele-
003155 ments of ",
003160      AT (18,08),
003165 "          a sentence (e.g. in, on, by, to, since) JM
003170 ",
003175      AT (19,03),  FAC(HEX(84)),    P$(11)       , CH(03),
003180      AT (19,08),
003185 "Pronoun .....Word that may be used as a substitute fo-
003190 r a noun",
003195      AT (20,08),
003200 "          (e.g. I, you, she, what,....)", JM
003205      AT (21,03),  FAC(HEX(84)),    P$(12)       , CH(03),
003210      AT (21,08),
003215 "Verb .....Denotes an action or occurrence",
003220      AT (23,20),
003225 "Press",
003230      AT (23,26),  FAC(HEX(84)),    PF1$        , CH(03),
003235      AT (23,30),
003240 "to return to the previous screen",
003245 KEYS(BIN(1)&BIN(17)) RT
003250 RETURN RT
003255 , RT
003260 PRINT SCREEN INIT: FLAG=1 :ER$=" ":E$=HEX(8D) RT
003265 PRINT SCREEN: RT
003270 ACCEPT RT
003275      AT (01,03), RT
003280 "CRES: Root Word Expansion Subsystem (Print, Display or Merge E RT
003285 .R.W. Option)", RT
003290      AT (02,03), RT
003295 "===== RT
003300 =====", RT
003305      AT (04,03), RT
003310 "Since you have completed the root word expansion list, please en RT
003315 ter the", RT
003320      AT (05,03), RT
003325 "appropriate", RT
003330      AT (05,15),  FAC(HEX(85)),    PFKEY$      , CH(06), RT
003335      AT (05,22), RT
003340 " corresponding to one of the following operations:", RT
003345      AT (07,10),  FAC(E$),        ER$          , CH(65), RT
003350      AT (09,17),  FAC(HEX(A5)),    PF$          , CH(02), RT
003355      AT (09,24),  FAC(HEX(AC)),    OPERATIONS$ , CH(27), RT
003360      AT (10,17),  FAC(HEX(84)),    A1$          , RT
003365      AT (10,24), RT
003370 "Display root word expansions", RT

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 13:41 11/13/84 Page 13  
Input File is CRES@ERW in Library CRES@SRC on Volume CRES02

```

003375      AT (11,17),  FAC(HEX(84)),      A2$          ,      : RT
003380      AT (11,24),                                     : RT
003385 "Print listing of root word expansions",          : RT
003390      AT (12,17),  FAC(HEX(84)),      A3$          ,      : RT
003395      AT (12,24),                                     : RT
003400 "Merge root words into",                            : RT
003405      AT (12,46),  FAC(HEX(85)),      FILE$        , CH(08), : RT
003410      AT (14,17),  FAC(HEX(84)),      A15$         ,      : RT
003415      AT (14,24),                                     : RT
003420 "Add or modify root word expansions before merging lists", : RT
003425      AT (15,17),  FAC(HEX(84)),      A16$         ,      : RT
003430      AT (15,24),                                     : RT
003435 "Exit the program without merging expansions",      : RT
003440      AT (18,03),                                     : RT
003445 "This optional subtitle may be used to identify a root word expan: RT
003450 sion list",                                           : RT
003455      AT (19,03),                                     : RT
003460 "that has been queued to the printer.",              : RT
003465      AT (21,14),                                     : RT
003470 "SUBTITLE:",                                           : RT
003475      AT (21,27),  FAC(HEX(81)),      SUB1$        , CH(23), : RT
003480      AT (22,27),  FAC(HEX(81)),      SUB2$        , CH(23), : RT
003485 KEYS(HEX(000111021203130F1F1020)), KEY(OPTION), : RT
003490 ON(BIN(16)&BIN(32)) GOTO SCRATCH_FILE,SCRATCH_FILE : RT
003495                                     : RT
003500 ER$=" ":E$=HEX(8C):WORD$,PRIO$PC$=" " : RT
003505 IF OPTION=1 OR OPTION =17 THEN VIEW : RT
003510 IF OPTION=2 OR OPTION =18 THEN VIEW : RT
003515 IF OPTION=3 OR OPTION =19 THEN MERGE : RT
003520 IF OPTION=15 OR OPTION=31 THEN MODIFY : RT
003525 GOTO PRINT_SCREEN : RT
003530                                     : RT
003535 MODIFY: : RT
003540 OPERATION$      ="OPERATION":ROOTWD$=" " : RT
003545 ACCEPT : RT
003550      AT (01,03),                                     : RT
003555 "CRES: Root Word Expansion Subsystem (Modify Expa: RT
003560 nsion Screen)", : RT
003565      AT (02,03),                                     : RT
003570 "===== : RT
003575 =====", : RT
003580      AT (04,03),                                     : RT
003585 " Please enter the word you want to modify and select the appropr: RT
003590 iate", : RT
003595      AT (04,72),  FAC(HEX(85)),      PFKEY$        , CH(06), : RT
003600      AT (04,79),                                     : RT
003605 " ", : RT
003610      AT (08,28),                                     : RT
003615 "Word =", : RT
003620      AT (08,35),  FAC(HEX(81)),      WORD$          , CH(20), : RT
003625      AT (10,25),  FAC(E$),          ER$            , CH(55), : RT
003630      AT (12,29),  FAC(HEX(A5)),      PF$            , CH(02), : RT
003635      AT (12,36),  FAC(HEX(AD)),      OPERATION$    , CH(15), : RT
003640      AT (13,29),  FAC(HEX(84)),      A1$            ,      : RT
003645      AT (13,36),                                     : RT
003650 "Add an Expansion", : RT
003655      AT (14,29),  FAC(HEX(84)),      A2$          ,      : RT

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 13:41 11/13/84 Page 14  
Input File is CRES@ERW in Library CRES@SRC on Volume CRES02

```

003660      AT (14,36),                                : RT
003665 "Delete the Word",                              : RT
003670      AT (15,29),      FAC(HEX(84)),      A3$      ,      : RT
003675      AT (15,36),                                : RT
003680 "Modify the Word",                              : RT
003685      AT (17,29),      FAC(HEX(84)),      A15$      ,      : RT
003690      AT (17,36),                                : RT
003695 "End Modifications",                          : RT
003700      AT (18,29),      FAC(HEX(84)),      A16$      ,      : RT
003705      AT (18,36),                                : RT
003710 "Return to the Master Menu",                  : RT
003715 KEYS(HEX(000111021203130F1F1020)), KEY(OPTION), : RT
003720 ON(BIN(15)&BIN(31)&BIN(16)&BIN(32))GOTO PRINT_SCREEN_INIT, : RT
003725      PRINT_SCREEN_INIT,SCRATCH_FILE,SCRATCH_FILE : RT
003730      : RT
003735 ER$=" ":E$=HEX(8C)                             : RT
003740      : RT
003745 FOR I= 1 TO LEN(WORD$)                          : RT
003750 IF POS("ABCDEFGHJKLMNOPQRSTUVWXYZ'-'=STR(WORD$,I,1))>0 THEN : JM
003755 3770      : RT
003760 ER$= "Illegal character found in word - please respecify." : RT
003765 E$=HEX(94):GOTO MODIFY                          : RT
003770 NEXT I      : RT
003775      : RT
003780 READ #2, HOLD, KEY=WORD$, IOBUFF$, BOD GOTO 3825 : RT
003785 IF OPTION= 2 OR OPTION= 18 THEN 3810              : RT
003790 IF OPTION= 3 OR OPTION= 19 THEN 3895              : RT
003795 ER$=" " & WORD$& " already exists in the list.":E$=HEX(94) : JM
003800 GOTO MODIFY      : RT
003805      : RT
003810 DELETE #2: ER$=" " & WORD$& " has been deleted.": E$=HEX(8D) : RT
003815 WORD$=" ": GOTO MODIFY      : RT
003820      : RT
003825 IF OPTION<>1 AND OPTION<>17 THEN ER1              : RT
003830 PRTO$PC$,PPART$,NEGS$,TOBES$,ROOT$,MTW$=" "      : RT
003835 GOSUB ATTRIBUTES      : RT
003840 IF OPTION=2 OR OPTION=18 THEN 3880              : RT
003845      : RT
003850      $PACK (F=F1$) IOBUFF$ FROM WORD$, PRTO$PC$, PPART$, : RT
003855      NEGS$,TOBES$,ROOT$,MTW$                      : RT
003860      WRITE #2, IOBUFF$, DATA GOTO IOERR2          : RT
003865      ER$=WORD$& " has been inserted.":E$=HEX(8C) : JM
003870      GOTO MODIFY      : RT
003875      : RT
003880 ER$=WORD$& " has been deleted from the expansions." : RT
003885 GOTO MODIFY      : RT
003890      : RT
003895 $UNPACK(F=F1$) IOBUFF$ TO WORD$, PRTO$PC$,PPART$,NEGS$,TOBES$, : RT
003900      ROOT$,MTW$                      : RT
003905 GOSUB ATTRIBUTES      : RT
003910 IF OPTION=2 OR OPTION=18 THEN 3810              : RT
003915      $PACK (F=F1$) IOBUFF$ FROM WORD$, PRTO$PC$, PPART$, : RT
003920      NEGS$,TOBES$,ROOT$,MTW$                      : RT
003925 REWRITE #2,IOBUFF$, DATA GOTO IOERR2          : RT
003930 ER$=" " & WORD$& " has been updated.": E$=HEX(8D) : RT
003935 GOTO MODIFY      : RT
003940      : RT

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 13:41 11/13/84 Page 15  
Input File is CRES@ERW in Library CRES@SRC on Volume CRES02

```

003945 ERL: ER$="      " &WORD$&" does not exist in this list."      RT
003950      ES=HEX(94): GOTO MODIFY      RT
003955      RT
003960 VIEW:      RT
003965 /* Now it's time to generate our print file name */      RT
003970      RT
003975      LPP%=50      RT
003980      DAT$= DATE      RT
003985      D$= STR(DAT$,3,2)&"/"&STR(DAT$,5,2)&"/"&STR(DAT$,1,2)      RT
003990      T$= STR(TIME,1,2)&":"&STR(TIME,3,2)      RT
003995      RT
004000      PFILE$="ROOT##"      RT
004005      I%=0      RT
004010 NEXT14: CONVERT I% TO STR(PFILE$,7,2),PIC(00)      RT
004015      I%=I%+1      RT
004020      CALL "LIMITS" (PFILE$,SLIB$,SVOL$,CODE$,ER$)      RT
004025      IF CODE$="Y" THEN NEXT14      RT
004030 DISPLAY      : RT
004035      AT (01,03),      : RT
004040 "CRES: Expanded Root Word Print Subsystem",      : RT
004045      AT (01,61),      : RT
004050 "(Generation Screen)",      : RT
004055      AT (02,03),      : RT
004060 "=====      : RT
004065 =====",      : RT
004070      AT (12,23),      : RT
004075 "Output file now being generated!"      RT
004080 OPEN NOGETPARM #3,OUTPUT,FILE=PFILE$,LIBRARY=SLIB$,      : RT
004085      VOLUME=SVOL$,SPACE=15000      RT
004090      GOSUB TTL PGE      RT
004095      STARTPH$ = ALL(HEX(00))      RT
004100      PAGE% = 1      RT
004105      GOSUB FIRST      RT
004110      RECS% = 0      RT
004115      READ #2, KEY)=STARTPH$,IOBUFF$,EOD GOTO HOME      RT
004120 KNOT: PSN= POS("JVCJNOInAPad"= STR(IOBUFF$,21,1))      RT
004125      IF PSN=0 THEN PSN=14      RT
004130      IF PSN=14 THEN AS(14)=STR(IOBUFF$,21,1)      RT
004135      IF OPTION=1 OR OPTION=17 THEN DISPFIL      RT
004140      WRITE #3, USING IMG6,C$, " ",STR(IOBUFF$,1,20),      : RT
004145      AS(PSN),STR(IOBUFF$,22,1),STR(IOBUFF$,24,1),      : RT
004150      STR(IOBUFF$,23,1),STR(IOBUFF$,27,1)      RT
004155      IF(STR(IOBUFF$,26,1)<>"*")THEN AGAN      RT
004160      C$=HEX(0000)      RT
004165      WRITE #3, USING IMG5, C$,STR(IOBUFF$,1,20)      RT
004170      WRITE #3, USING IMG9, C$,STR(IOBUFF$,1,20)      RT
004175      C$=HEX(0001)      RT
004180      GOTO AGAN      RT
004185 DISPFIL: IF STR(IOBUFF$,26,1)="*" THEN STR(IOBUFF$,26,1)="+"      RT
004190      WRITE #3, USING IMG6,C$,STR(IOBUFF$,26,1),STR(IOBUFF$,1,20),: RT
004195      AS(PSN),STR(IOBUFF$,22,1),STR(IOBUFF$,24,1),      : RT
004200      STR(IOBUFF$,23,1),STR(IOBUFF$,27,1)      RT
004205 AGAN: IF RECS%=LPP% THEN RECS%=1      RT
004210      READ #2, IOBUFF$,EOD GOTO HOME      RT
004215      IF RECS% = 1 AND PAGE%<?1 THEN GOSUB HEADER      RT
004220      RECS%=RECS%+1      RT
004225      GOTO KNOT      RT

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 13:41 11/13/84 Page 16  
Input File is CRES00ERW in Library CRES0SRC on Volume CRES02

```

004230 HOME: IF RECS%<2 THEN RECS%=LPP% RT
004235 WRITE #3, USING TIT, C$ RT
004240 RECS%=RECS%+2 RT
004245 WRITE #3, USING DG, C$ RT
004250 FOR I = RECS% TO LPP% RT
004255 WRITE #3, USING TIT, C$ RT
004260 NEXT I RT
004265 IF OPTION=2 OR OPTION=18 THEN HER1 RT
004270 WRITE #3, USING IMG8, C$ RT
004275 GOTO HER RT
004280 HER1: WRITE #3, USING ROT, C$ RT
004285 WRITE #3, USING ROT1, HEX(0000) RT
004290 WRITE #3, USING ROT1, HEX(0000) RT
004295 HER: CLOSE #3 RT
004300 IF OPTION=2 OR OPTION=18 THEN MESS RT
004305 CLOSE WS RT
004310 MSG2$ = "CANCEL Processing" RT
004315 MSK$=HEX(00) RT
004320 MSG$ = " " RT
004325 CALL "LINK" ADDR("DISPFILE", "P", RL18$, RVOL$, 3$, PFILE$, : RT
004330 SL18$, SVOL$, "C", MSG2$, 27$, "H", MSK$, CMGS$, 255$, C$, R$) RT
004335 ER$ = " " RT
004340 IF C%<>0 OR R%<>0 : RT
004345 THEN ER$ = STR(CMSG$, 141) RT
004350 GOTO PRINT_SCREEN RT
004355 RT
004360 MESS: ER$=" File "&PFILE$&" in Library "& SL18$ &" on " & : RT
004365 SVOL$ &" generated.": E$=HEX(84) RT
004370 GOTO PRINT_SCREEN RT
004375 RT
004380 TTL_PGE: RT
004385 WRITE #3, USING TIT, HEX(8001) RT
004390 WRITE #3, USING TIT, HEX(0015) RT
004395 C$ = HEX(0001) RT
004400 WRITE #3, USING FIT1, C$ RT
004405 WRITE #3, USING FIT2, C$ RT
004410 WRITE #3, USING FIT0, C$, D$, T$ RT
004415 WRITE #3, USING FIT3, C$ RT
004420 WRITE #3, USING FIT2, C$ RT
004425 WRITE #3, USING FIT1, C$ RT
004430 WRITE #3, USING FIT2, C$ RT
004435 IF SUB1$ = " " THEN GERRY RT
004440 WRITE #3, USING FIT4, C$, SUB1$ RT
004445 IF SUB2$<>" " THEN WRITE #3, USING FIT10, C$, SUB2$ RT
004450 WRITE #3, USING FIT2, C$ RT
004455 GERRY: WRITE #3, USING FIT5, C$, DFIL$ RT
004460 WRITE #3, USING FIT2, C$ RT
004465 WRITE #3, USING FIT6, C$, DL18$ RT
004470 WRITE #3, USING FIT2, C$ RT
004475 WRITE #3, USING FIT7, C$, DVOL$ RT
004480 WRITE #3, USING FIT2, C$ RT
004485 WRITE #3, USING FIT1, C$ RT
004490 RETURN RT
004495 RT
004500 HEADER: RT
004505 PAGE%=PAGE%+1 RT
004510 IF OPTION=2 OR OPTION=18 THEN FRST1 RT

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 13:41 11/13/84 Page 17  
Input File is CRES00ERW in Library CRES00SRC on Volume CRES02

```

004515      WRITE #3, USING FIT, C$      RT
004520      WRITE #3, USING IMG8, C$      RT
004525      GOTO FIRST                    RT
004530 FIRST: WRITE #3, USING ROT, C$      RT
004535      WRITE #3, USING ROT1, HEX(0000) RT
004540      WRITE #3, USING ROT1, HEX(0000) RT
004545 FIRST: WRITE #3, USING FIT, HEX(8001) RT
004550      IF SUB1$ = " "                  ! RT
004555      THEN WRITE #3, USING IMG, C$, D$, T$, PAGE$ ! RT
004560      ELSE GOTO FOUR                  RT
004565      GOTO ALRITE                    RT
004570 FOUR:  WRITE #3, USING IMG1, C$, SUB1$, D$, T$, PAGE$ RT
004575      IF SUB2$ <> " "                  ! RT
004580      THEN WRITE #3, USING IMG7, C$, SUB2$      RT
004585 ALRITE: WRITE #3, USING IMG2, C$, DPIL$, DLIB$, DVOL$ RT
004590      WRITE #3, USING IMG3, C$      RT
004595      WRITE #3, USING IMG4, C$      RT
004600      WRITE #3, USING IMG5, C$      RT
004605      WRITE #3, USING IMG3, C$      RT
004610      RETURN                        RT
004615      RT
004620 FIT0:###      :      #####      ! RT
004625      #####      !      RT
004630 FIT:###      RT
004635 DG:###      !!! END OF LISTING !!! RT
004640 FIT1:###      !-----!      RT
004645 -----!      RT
004650 FIT2:###      :      ! RT
004655      :      ! RT
004660 FIT3:###      :      ***** ROOT WORD EXPANSION ***** RT
004665 *      :      RT
004670 FIT4:###      :      TITLE OF LIST: ##### RT
004675 #####      :      RT
004680 FIT5:###      :      , EXPANSION FILE: ##### ! RT
004685      :      RT
004690 FIT6:###      :      EXPANSION LIBRARY: ##### ! RT
004695      :      RT
004700 FIT7:###      :      EXPANSION VOLUME: ##### ! RT
004705      :      RT
004710 FIT10:###      :      ##### RT
004715 #####      :      RT
004720 IMG:###      #####      ! RT
004725      PAGE #####      RT
004730 IMG1:###      TITLE OF LIST : #####      ! RT
004735 #####      PAGE #####      RT
004740 IMG7:###      ##### RT
004745 IMG2:###      Input File is ##### in Library ##### on Volume ### RT
004750 ###      RT
004755 IMG3:###      =====! RT
004760 ===== RT
004765 IMG4:###      WORD      Part of      Past      To Be Ne! RT
004770 gative      Mand Tech      RT
004775 IMG5:###      Speech      Participle      ! RT
004780      word      RT
004785 IMG6:###      # #####      #      #      ! RT
004790      #      RT
004795 IMG9:###      ##### RT

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 13:41 11/13/84 Page 18  
Input File is CRES@ERW in Library CRES@SRC on Volume CRES02

```

004800 IMG8:### + = ROOT WORD RT
004805 ROT:### BOLDFACE = Root Word RT
004810 ROT1:### BOLDFACE RT
004815 RT
004820 MERGE: FLAG=0 RT
004825 DISPLAY : RT
004830 AT (01,03), : RT
004835 "CRES: Expanded Root Word Merge Subsystem", : RT
004840 AT (01,61), : RT
004845 "(Generation Screen)", : RT
004850 AT (02,03), : RT
004855 "===== RT
004860 =====", : RT
004865 AT (12,23), : RT
004870 "Merging is now taking place!" RT
004875 IF OPEN%(2)=1 THEN CLOSE #2 RT
004880 OPEN NOGETPARM #2, IO, FILE=DFIL$, LIBRARY=DLIB$, VOLUME=DVOL$ RT
004885 OPEN%(2)=1 RT
004890 READ #2, IOBUFF$, EOD GOTO 4940 RT
004895 READ #1, HOLD, KEY=STR(IOBUFF$,20), BUFF$, EOD GOTO INSERT RT
004900 REWRITE #1, IOBUFF$ RT
004905 GOTO 4890 RT
004910 INSERT: RT
004915 WRITE #1, IOBUFF$ RT
004920 GOTO 4890 RT
004925 RT
004926 SCRATCH FILE: RT
004930 ER$= "Before returning to the Master Menu, consider the following! RT
004935 deletion." :GOTO 4945 RT
004940 FLAG=0: ER$=" Merge has been successfully completed!" RT
004945 E$=HEX(84) RT
004955 ACCEPT : RT
004960 AT (01,03), : RT
004965 "CRES: Root Word Expansion Subsystem (Merge Compl RT
004970 etion Screen)", : RT
004975 AT (02,03), : RT
004980 "===== RT
004985 =====", : RT
004990 AT (05,04), FAC(E$), ER$, CH(70), RT
004995 AT (08,04), RT
005000 "Press", RT
005005 AT (08,10), FAC(HEX(84)), PF2$ , CH(03), RT
005010 AT (08,14), RT
005015 ", if you want to delete the work file consisting of the root wor RT
005020 d", RT
005025 AT (09,04), RT
005030 "list you have just created. IF", RT
005035 AT (09,36), FAC(HEX(84)), DFIL$ , CH(06), RT
005040 AT (09,43), RT
005045 "is not deleted from", RT
005050 AT (09,63), FAC(HEX(85)), DVOL$ , CH(06), RT
005055 AT (09,70), RT
005060 ", then", RT
005065 AT (10,04), RT
005070 "the system utility 'DISPLAY' may be used at a later date to view RT
005075 this list", RT
005080 AT (11,04), RT

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 13:41 11/13/84 Page 19  
Input File is CRES@ERW in Library CRES@SRC on Volume CRES02

```

005085 "again. However, it is advised to delete these files when possible" RT
005090 "le to avoid", RT
005095 AT (12,04), RT
005100 "unnecessary accumulation of expanded root word files on disk.", RT
005101 AT (16,15), FAC(E1$), E1$, RT
005105 AT (22,17), RT
005110 "Press", RT
005115 AT (22,23), FAC(HEX(84)), PF15$ , CH(04), RT
005120 AT (22,28), RT
005125 "to return to the file selection screen", RT
005130 AT (23,22), RT
005135 "Press", RT
005140 AT (23,28), FAC(HEX(84)), PF16$ , CH(04), RT
005145 AT (23,33), RT
005150 "to return to the Master Menu", RT
005155 KEYS(BIN(2)&BIN(18)&BIN(15)&BIN(31)&BIN(16)&BIN(32)), RT
005160 ON (BIN(2)&BIN(18)&BIN(16)&BIN(32)) GOTO SCRAT,SCRAT,EXIT,EXIT RT
005165 RT
005170 ER$=" ": E$=HEX(8C): FILE$=" ":GOTO FILMWU RT
005175 / RT
005180 SCRAT: IF OPEN$(2)=1 THEN CLOSE #2:OPEN$(2)=0 RT
005185 CALL "LIMITS" (DFIL$,DLIB$,DVOL$,CODE$) RT
005190 IF CODE$="Y" THEN 5205 RT
005195 E1$="No work file was created on this run" RT
005200 E1$=HEX(84): ER$=" ": E$=HEX(8C): GOTO 4955 RT
005205 CALL "SCRATCH" ADDR("F",DFIL$,DLIB$,DVOL$,B") RT
005210 E1$= DFIL$&" in "&DLIB$&" on "&DVOL$&" has been deleted." RT
005215 E1$=HEX(84) :ER$=" ":E$=HEX(8C) RT
005220 GOTO 4955 RT
005225 RT
005230 EXIT: IF OPEN$(1)=1 THEN CLOSE #1: IF OPEN$(2)=1 THEN CLOSE #2 RT
005235 IF OPEN$(3)=1 THEN CLOSE #3 : END RT
005240 RT
005245 /* THIS FUNCTION PRINTS OUT ERROR MESSAGES AND */ RT
005250 /* SETS UP FLASHING FAC'S IN APPROPRIATE PLACES */ RT
005255 DEF FN'7(ER$,B$,C$,D$) RT
005260 PRINT RT
005265 E$=HEX(94) RT
005270 F$=B$ RT
005275 L$=C$ RT
005280 V$=D$ RT
005285 RETURN RT
005290 RT

```

Technical Report 86-001

THIS PAGE INTENTIONALLY LEFT BLANK.

## CRES ANALYSIS SUBSYSTEM

### User Default Set-Up (KIKOFF)

KIKOFF in library CRES@SRC is the BASIC language program to set up and supply default values to the CRES Analysis program. The program checks for the existence of a defaults file for the particular user. (The file name is formed by concatenating the user ID and the string "DFLTS", and the file is stored in library CRES@DAT.) If the file exists, the defaults are set accordingly. If the file does not exist, a flag is set to create a defaults file for that user at the end of the analysis run. The object language version is KIKOFF in library CRES@SAV.

A complete program listing follows.

Technical Report 86-001

THIS PAGE INTENTIONALLY LEFT BLANK.

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 13:12 09/17/84 Page 1  
Input File is KICKOFF in Library CRES0SRC on Volume CRES00

```

000010 /***** JM
000030 /* JM
000050 /* SUBROUTINE KICKOFF SETS UP THE USER DEFAULTS FOR THE JM
000070 /* ANALYSIS. THIS BLOCK CHECKS IF A DEFAULT FILE EXISTS FOR JM
000090 /* THE PARTICULAR USER. IF THE FILE EXISTS, THE DEFAULTS ARE JM
000110 /* FETCHED AND USED. IF THE FILE DOES NOT EXIST, THE FILE IS JM
000130 /* CREATED AND DEFAULTS ARE SUPPLIED. JM
000150 /* JM
000170 /***** JM
000190 JM
000210 SUB "KICKOFF" ADDR(RGL$,PREP CNT$,PASSV$,LNG SEN$,LNG PAR$,LNGWDS$: JM
000230 ,LNGSNT$,NEG$,UNCOM$,NUM SUBZ$,INT ANL$,NP$,ADVFN$,PERS SENT$, JM
000250 LNG CLS$,TRNK PV$,TECFIL$(),TECLIB$(),TECVOL$(),RETS$,INLIB$, JM
000270 INVOL$) JM
000290 DIM RGL$1,PREP CNT$1,RET$1,INVOL$6,INLIB$08 JM
000310 DIM PASSV$1,LNG SEN$1,LNG PAR$1,LNGWDS$3,LNGSNT$2,NEG$1,UNCOM$1 JM
000330 DIM NUM SUBZ$1,INT ANL$1,NP$2,ADVFN$1,PERS SENT$1,LNG CLS$1 JM
000350 DIM TRNK PV$1,TECFIL$(7)08,TECLIB$(7)08,TECVOL$(7)06 JM
000370 DIM F1$10,BUFF$190,F2$32,F3$14,F4$14,BUFF1$20,BUFF2$52,BUFF3$52 JM
000390 F1$=HEX(A0740010A034A034A032) JM
000410 F2$=HEX(A001A001A001A001A001A003A002A001A001A001A002A001A001A: JM
000430 001A001) JM
000450 F3$=HEX(A008A008A006A008A008A006A008) JM
000470 F4$=HEX(A008A006A008A008A006A008A008) JM
000490 F5$=HEX(A006A008A008A006A008A008A006) JM
000510 SELECT #1, "SETDFLT", CONSEC, RECSIZE=190, IOERR GOTO PROB1 JM
000530 SELECT #2, "GETDFLT", CONSEC, RECSIZE=190, IOERR GOTO PROB2 JM
000550 CALL "EXTRACT" ADDR("ID", USER$) /* FETCH THE USER'S ID */ JM
000570 FILO$=USER$ & "DFLT" JM
000590 /* SEE IF THE DEFAULTS FILE EXISTS FOR THIS USER */ JM
000610 JM
000630 CALL "LIMITS"(FILO$,INLIB$,INVOL$,RETC$,MESG$) JM
000650 IF RETC$="Y" THEN FETCH JM
000670 JM
000690 /* IF HERE, WE NEED TO GENERATE A DEFAULTS FILE */ JM
000710 /* WE NEED THE ANALYSIS SCREENS TO ASK FOR DEFAULTS, */ JM
000730 /* SO SET RETURN CODE AND GO BACK TO ANALYSIS PROGRAM */ JM
000750 JM
000770 OPEN NOGETPARM #1, OUTPUT, SPACE=1, FILE=FILO$,LIBRARY= : JM
000790 INLIB$, VOLUME=INVOL$ JM
000810 BUFF1$ = RGL$ & PREP CNT$ & PASSV$ & LNG SEN$ & LNG PAR$ & : JM
000830 LNGWDS$ & LNGSNT$ & NEG$ & UNCOM$ & NUM SUBZ$ & INT ANL$ & : JM
000850 NP$ & ADVFN$ & PERS SENT$ & LNG CLS$ & TRNK PV$ JM
000870 BUFF2$ = STR(TECFIL$(1),1,8) & STR(TECLIB$(1),1,8) & : JM
000890 STR(TECVOL$(1),1,6) & STR(TECFIL$(2),1,8) & STR(TECLIB$(2),1,8) : JM
000910 & STR(TECVOL$(2),1,6) & STR(TECFIL$(3),1,8) JM
000930 BUFF3$ = STR(TECLIB$(3),1,8) & STR(TECVOL$(3),1,6) & : JM
000950 STR(TECFIL$(4),1,8) & STR(TECLIB$(4),1,8) & STR(TECVOL$(4),1,6) : JM
000970 & STR(TECFIL$(5),1,8) & STR(TECLIB$(5),1,8) JM
000990 BUFF4$ = STR(TECVOL$(5),1,6) & STR(TECFIL$(6),1,8) & : JM
001010 STR(TECLIB$(6),1,8) & STR(TECVOL$(6),1,6) & STR(TECFIL$(7),1,8) : JM
001030 & STR(TECLIB$(7),1,8) & STR(TECVOL$(7),1,6) JM
001050 $PACK (F=FILO$) BUFF$ FROM BUFF1$, BUFF2$, BUFF3$, BUFF4$ JM
001070 WRITE #1, BUFF$ JM

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 13:12 09/17/84 Page 2  
Input File is KICKOFF in Library CRES@SRC on Volume CRES00

```

000970      CLOSE #1                                JM
000990      RET$="L" /* SO WE'LL LOAD DEFAULTS AFTER USER SETS THEM */ JM
001010      GOTO IT                                  /* RETURN */ JM
001030      JM
001050  FETCH:                                     JM
001060      IF RET$="L" THEN MISS                     JM
001070      OPEN NOGETPARM #2, IO, FILE=FILO$,LIBRARY = INLIB$, : JM
001090      VOLUME=INVOL$                             JM
001110      READ #2, BUFF$                            JM
001130      $UNPACK (F=F1$) BUFF$ TO BUFF1$,BUFF2$,BUFF3$,BUFF4$ JM
001132      $UNPACK (F=F2$) BUFF1$ TO RGL$,PREP CNT$,PASSV$,LNG SEN$, : JM
001134      LNG PAR$,LNGWDS$,LNGSNT$,NEG$,UNCOM$,NUM SUBZ$,INT ANL$, : JM
001136      NP$,ADVFN$,PERS SENT$,LNG CLS$,TRNK PV$ JM
001138      $UNPACK (F=F3$) BUFF2$ TO TECFIL$(1),TECLIB$(1),TECVOL$(1),: JM
001140      TECFIL$(2),TECLIB$(2),TECVOL$(2),TECFIL$(3) JM
001144      $UNPACK (F=F4$) BUFF3$ TO TECLIB$(3),TECVOL$(3),TECFIL$(4),: JM
001146      TECLIB$(4),TECVOL$(4),TECFIL$(5),TECLIB$(5) JM
001147      $UNPACK (F=F5$) BUFF4$ TO TECVOL$(5),TECFIL$(6),TECLIB$(6),: JM
001148      TECVOL$(6),TECFIL$(7),TECLIB$(7),TECVOL$(7) JM
001270      CLOSE #2                                JM
001290      GOTO IT                                  /* RETURN */ JM
001310      JM
001330  MISS:                                     JM
001350      OPEN NOGETPARM #2, IO, FILE=FILO$,LIBRARY = INLIB$, : JM
001370      VOLUME=INVOL$                             JM
001380      BUFF1$ = RGL$ & PREP CNT$ & PASSV$ & LNG SEN$ & LNG PAR$ & : JM
001390      LNGWDS$ & LNGSNT$ & NEG$ & UNCOM$ & NUM SUBZ$ & INT ANL$ & : JM
001400      NP$ & ADVFN$ & PERS SENT$ & LNG CLS$ & TRNK PV$ JM
001410      BUFF2$ = STR(TECFIL$(7),1,8) & STR(TECLIB$(1),1,8) & : JM
001415      STR(TECVOL$(1),1,6) & STR(TECFIL$(2),1,8) & STR(TECLIB$(2),1,8) : JM
001420      & STR(TECVOL$(2),1,6) & STR(TECFIL$(3),1,8) JM
001430      BUFF3$ = STR(TECLIB$(3),1,8) & STR(TECVOL$(3),1,6) & : JM
001435      STR(TECFIL$(4),1,8) & STR(TECLIB$(4),1,8) & STR(TECVOL$(4),1,6) : JM
001440      & STR(TECFIL$(5),1,8) & STR(TECLIB$(5),1,8) JM
001450      BUFF4$ = STR(TECVOL$(5),1,6) & STR(TECFIL$(6),1,8) & : JM
001455      STR(TECLIB$(6),1,8) & STR(TECVOL$(6),1,6) & STR(TECFIL$(7),1,8) : JM
001460      & STR(TECLIB$(7),1,8) & STR(TECVOL$(7),1,6) JM
001470      $PACK (F=F1$) BUFF$ FROM BUFF1$, BUFF2$, BUFF3$, BUFF4$ JM
001475      READ #2, HOLD, BUFF9$                    JM
001530      REWRITE #2, BUFF$                        JM
001550      CLOSE #2                                  JM
001570      GOTO IT                                  JM
001590  PROB1:                                     JM
001610      PRINT AT(23,1), "I/O ERROR ON DEFAULTS FILE, #1. FILE";: JM
001630      " STATUS IS : "; FS(#1)                   JM
001650      STOP "PRESS ENTER TO CONTINUE"             JM
001670      CLOSE #1                                    JM
001690      GOTO IT                                     JM
001710      JM
001730  PROB2: PRINT AT(23,1), "I/O ERROR ON DEFAULTS FILE, #2" : JM
001750      ";. FILE STATUS IS : ";FS(#2)               JM
001770      STOP "PRESS ENTER TO CONTINUE"             JM
001790      CLOSE #2                                    JM
001810      JM
001830  IT:    END                                  JM

```

Document and Word List Selection (SCRENZ)

SCRENZ in library CRES@SRC is the BASIC language program to allow user selection of the document to be analyzed and the names of the Common Word List, Word Substitution List, and Phrase Substitution List to be used by the CRES Analysis program. Default values are supplied, and may be changed by the user if desired. The program verifies that the document and lists selected actually exist. The object language version is SCRENZ in library CRES@SAV.

A complete program listing follows.

Technical Report 86-001

THIS PAGE INTENTIONALLY LEFT BLANK.

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 15:36 11/13/84 Page 1  
Input File is SCREEN in Library CRES@SRC on Volume CRES02

```

000010 SUB "SCREENZ" (COMFIL$,COMLIB$,COMVOL$,WRDFIL$,WRDLIB$,WRDVOL$, : JM
000020 PHRFIL$,PHRLIB$,PHRVOL$,DOCFIL$,DOCLIB$,DCVOL$,PAG_TITL$, : JM
000030 LIB$,REG$,INLIB$,INVOL$) JM
000040
000050 SELECT #1,"CMNWDLST",VARC,INDEXED,RECSIZE=40, :
000060 KEYPOS=01, KEYLEN=20, IOERR GOTO IO_ERR1
000070 SELECT #2,"WRDSBLST",VARC,INDEXED,RECSIZE=80, :
000080 KEYPOS=01, KEYLEN=20, IOERR GOTO IO_ERR2
000090 SELECT #3,"PHRSBLST",VARC,INDEXED,RECSIZE=70, :
000100 KEYPOS=01, KEYLEN=20, IOERR GOTO IO_ERR3
000120 DIM WRDFIL$8,WRDLIB$8,WRDVOL$6,FG$(6)01,LI3$1
000130 DIM PHRFIL$8,PHRLIB$8,PHRVOL$6
000140 DIM COMFIL$8,COMLIB$8,COMVOL$6,OPEN$(3)
000150 DIM AY$2,ES$65,FER$1,AS$8,B$16,C$24,D$22,BUFF$50 JM
000151 ES$=" " JM
000160 ENTER$="ENTER"
000170 PF$="PF16"
000171 MAT OPEN$ = ZER
000180 COMFIL$="CMNWDLST"
000190 PHRFIL$="PHRSBLST"
000200 WRDFIL$="WRDSBLST"
000220 DOCLIB$=" "
000230 * COMLIB$,PHRLIB$,WRDLIB$="CRES@DAT" JM
000240 * COMVOL$,PHRVOL$,WRDVOL$="FIXVOL" JM
000250 FG$()=ALL(HEX(81)) :FG$(4)=HEX(82) :FG$(5)=HEX(80)
000260 FER$=HEX(8C)
000270 AS$="Document"
000280 B$="Common Word List"
000290 C$="Phrase Substitution List"
000300 D$="Word Substitution List"
000310 SCREEN1: JM
000315 ACCEPT JM
000320 AT (01,03),
000330 "CRES: Document Analysis Subsystem (Document & Word Lists Sele
000340 ction Screen)",
000350 AT (02,03),
000360 "=====
000370 =====",
000380 AT (04,10),
000390 "Welcome to the Document Analysis Option. Please identify the",
000400 AT (05,10),
000410 "document to be analyzed and identify the Common Word List,",
000420 AT (06,10),
000430 "Phrase Substitution List, and the Word Substitution List ",
000440 AT (07,10),
000450 "to be used, if the defaults are not satisfactory. If desired,",
000460 AT (08,10),
000470 "input a title to be printed on each page.",
000480 AT (10,10),FAC(HEX(8C)),
000490 AS$,
000500 AT (10,19), ":",
000510 AT (11,12),
000520 "FILE :",
000530 AT (11,20), FAC(FG$(04)), DOCFIL$ , CH(04),
000540 AT (11,32),
000550 "LIBRARY :",
000560 AT (11,43), FAC(FG$(05)), LIB$ , CH(01),

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 15:36 11/13/84 Page 2  
Input File is SCREEN in Library CRES@SRC on Volume CRES02

```

000570      AT (11,55),
000580 "VOLUME :",
000590      AT (11,65), FAC(FG$(06)), DCVOL$      , CH(06),
000600      AT (13,10), FAC(HEX(8C)),
000610 B$, AT (13,27), ":",
000620      AT (14,12),
000630 "FILE :",
000640      AT (14,20), FAC(FG$(01)), COMFIL$      , CH(08),
000650      AT (14,32),
000660 "LIBRARY :",
000670      AT (14,43), FAC(FG$(01)), COMLIB$      , CH(08),
000680      AT (14,55),
000690 "VOLUME :",
000700      AT (14,65), FAC(FG$(01)), COMVOL$      , CH(06),
000710      AT (15,10), FAC(HEX(8C)),
000720 C$, AT (15,35), ":",
000730      AT (16,12),
000740 "FILE :",
000750      AT (16,20), FAC(FG$(02)), PHRFIL$      , CH(08),
000760      AT (16,32),
000770 "LIBRARY :",
000780      AT (16,43), FAC(FG$(02)), PHRLIB$      , CH(08),
000790      AT (16,55),
000800 "VOLUME :",
000810      AT (16,65), FAC(FG$(02)), PHRVOL$      , CH(06),
000820      AT (17,10), FAC(HEX(8C)),
000830 D$, AT (17,33), ":",
000840      AT (18,12),
000850 "FILE :",
000860      AT (18,20), FAC(FG$(03)), WRDFIL$      , CH(08),
000870      AT (18,32),
000880 "LIBRARY :",
000890      AT (18,43), FAC(FG$(03)), WRDLIB$      , CH(08),
000900      AT (18,55),
000910 "VOLUME :",
000920      AT (18,65), FAC(FG$(03)), WRDVOL$      , CH(06),
000930      AT (20,10),
000940 "Title:",
000950      AT (21,11), PAG_TITL$      , CH(60),
000960      AT (22,04),
000965      FAC(FER$), E$      , CH(65),
000970      AT (23,28),
000980 "Press",
000990      AT (23,34), FAC(HEX(84)), ENTER$,
001000      AT (23,40),
001010 "to continue",
001020      AT (24,18), "Press",
001030      AT (24,24), FAC(HEX(84)), PR$,
001040      AT (24,29),
001050 "to return to the CRES Master Menu",
001060 KEYS(HEX(001020)), KEY(RE$),
001070 ON (BIN(16)&BIN(32)) GOTO GONE,GONE
001080
001081 E$=" " :FER$=HEX(8C)
001090 FLAG$ = 0 : DOCLIB$="
001100
001110 /* SEE IF FILES EXIST THAT THE USER INPUT

```

JM  
JM

JM

\*/

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 15:36 11/13/84 Page 3  
Input File is SCREEN in Library CRES@SRC on Volume CRES02

```

001120
001130 IF LIB$ > "Z" THEN DOCLIB$ = "DOCMNT" & LIB$ & LIB$      :
001140 ELSE DOCLIB$ = "DOCMNT" & LIB$                          :
001150 TRAN (DOCLIB$,                                           :
001160 "AaBbCcDdEeFfGgHhIiJjKkLlMmNnOoPpQqRrSsTtUuVvWwXxYyZz")REPLACING
001170
001180 CALL "LIMITS"(DOCFIL$,DOCLIB$,DCVOL$,RC$,MSG$)
001190 IF RC$ = "Y" THEN LP
001200 FG$(4)=HEX(92) : FG$(5)=HEX(90) : FG$(6)=HEX(91)
001210 GOTO SCREEN1
001220
001230 LP: IF COMFIL$="CMNWDLST" AND COMLIB$=INLIB$ AND COMVOL$= : JM
001240 INVOL$ THEN NI JM
001250 IF COMFIL$=" " OR COMLIB$=" " OR COMVOL$=" " THEN NIS
001260 FLAG% = 1
001270 CALL "LIMITS"(COMFIL$,COMLIB$,COMVOL$,RC$,MSG$)
001280 IF RC$="Y" THEN NI
001290 NIS: FG$()=ALL(HEX(81)) :FG$(4)=HEX(82) :FG$(5)=HEX(80)
001300 FER$=HEX(94)
001310 ES$="Common Word List "& STR(COMFIL$,1,8) & " in " & :
001320 STR(COMLIB$,1,8) & " on "& STR(COMVOL$,1,6) & :
001330 " does not exist." :
001340 COMFIL$="CMNWDLST" :COMLIB$=INLIB$ :COMVOL$=INVOL$ JM
001350 FLAG%=0
001360 GOTO SCREEN1
001370
001380 NI: IF PHRFIL$="PHRSBLST" AND PHRLIB$=INLIB$ AND PHRVOL$= INVOL$ : JM
001390 THEN NI2 JM
001400 IF PHRFIL$=" " OR PHRLIB$=" " OR PHRVOL$=" " THEN NIS2
001410 FLAG% = FLAG%+2
001420 CALL "LIMITS"(PHRFIL$,PHRLIB$,PHRVOL$,RC$,MSG$)
001430 IF RC$="Y" THEN NI2
001440 NIS2: FG$()=ALL(HEX(81)) :FG$(4)=HEX(82) :FG$(5)=HEX(80)
001450 FER$=HEX(94)
001460 ES$="Phrase Sub. List "& STR(PHRFIL$,1,8) & " in " & :
001470 STR(PHRLIB$,1,8) & " on "& STR(PHRVOL$,1,6) & :
001480 " does not exist." :
001490 PHRFIL$="PHRSBLST" :PHRLIB$=INLIB$ :PHRVOL$=INVOL$ JM
001500 FLAG%=0
001510 GOTO SCREEN1
001520
001530 NI2:IF WRDFIL$="WRDSBLST" AND WRDLIB$=INLIB$ AND WRDVOL$=INVOL$ : JM
001540 THEN NI3 JM
001550 IF WRDFIL$=" " OR WRDLIB$=" " OR WRDVOL$=" " THEN NIS3
001560 FLAG%=FLAG%+4
001570 CALL "LIMITS"(WRDFIL$,WRDLIB$,WRDVOL$,RC$,MSG$)
001580 IF RC$="Y" THEN NI3
001590 NIS3: FG$()=ALL(HEX(81)) :FG$(4)=HEX(82) :FG$(5)=HEX(80)
001600 FER$=HEX(94)
001610 ES$="Word Sub. List "& STR(WRDFIL$,1,8) & " in " & :
001620 STR(WRDLIB$,1,8) & " on "& STR(WRDVOL$,1,6) & :
001630 " does not exist." JM
001640 WRDFIL$="WRDSBLST" :WRDLIB$=INLIB$ :WRDVOL$= INVOL$ JM
001650 FLAG%=0
001660 GOTO SCREEN1
001670 NI3: /* If WE COME HERE, WE KNOW THE THREE FILES EXIST. */ JM
001680 /* BUT ARE THEY THE KINDS OF FILES WE THINK THEY ARE? */

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 15:36 11/13/84 Page 4  
Input File is SCREEN in Library CRES0SRC on Volume CRES02

```

001690
001700 CALL "READFDR" ADDR(COMFIL$,COMLIB$,COMVOL$,0%,"RS",REC%,RET%)
001710 IF REC%<>40 THEN ER1
001711 OPEN%(1)=1
001720 OPEN NOGETPARM #1, SHARED, FILE=COMFIL$,
001730 LIBRARY=COMLIB$, VOLUME=COMVOL$
001740 READ #1, BUFF$
001750 IF STR(BUFF$,1,13)=" COMMON WORDS" THEN TRY
001760 ER1:ES="File "&STR(COMFIL$,1,8)&" in "&STR(COMLIB$,1,8)&" on "& ! JM
001770 STR(COMVOL$,1,6) &" is not a Common Word List."
001780 COMFIL$="CMNWDLS" :COMLIB$ = INLIB$ :COMVOL$ = INVOL$ JM
001790 FER$=HEX(94)
001800 IF OPEN%(1)=1 THEN CLOSE #1
001801 OPEN%(1)=0
001810 GOTO SCREEN1
001820
001830 TRY:CALL "READFDR" ADDR(WRDFIL$,WRDLIB$,WRDVOL$,0%,"RS",REC%,RET%)
001840 IF REC%<>80 THEN ER2
001841 OPEN%(2)=1
001850 OPEN NOGETPARM #2, SHARED, FILE=WRDFIL$,
001860 LIBRARY=WRDLIB$, VOLUME=WRDVOL$
001870 READ #2, BUFF$
001880 IF STR(BUFF$,1,18)=" WORD SUBSTITUTION" THEN AGN
001890 ER2:ES="File "&STR(WRDFIL$,1,8)&" in "&STR(WRDLIB$,1,8)&" on " & ! JM
001900 STR(WRDVOL$,1,6) &" is not a Word Sub. List." JM
001910 WRDFIL$="WRDSBLST" :WRDLIB$= INLIB$ :WRDVOL$= INVOL$ JM
001920 FER$=HEX(94)
001921 IF OPEN%(1)=1 THEN CLOSE #1
001926 IF OPEN%(2)=1 THEN CLOSE #2
001930 MAT OPEN% = ZER
001940 GOTO SCREEN1
001950
001960 AGN:CALL "READFDR" ADDR(PHRFIL$,PHRLIB$,PHRVOL$,0%,"RS",REC%,RET%)
001970 IF REC%<>70 THEN ER3
001971 OPEN%(3)=1
001980 OPEN NOGETPARM #3, SHARED, FILE=PHRFIL$,
001990 LIBRARY=PHRLIB$, VOLUME=PHRVOL$
002000 READ #3, BUFF$
002010 IF STR(BUFF$,1,20)=" PHRASE SUBSTITUTION" THEN AGAN1
002020 ER3:ES="File "&STR(PHRFIL$,1,8)&" in "&STR(PHRLIB$,1,8) &" on "& ! JM
002030 STR(PHRVOL$,1,6) &" is not a Phrase Sub. List." JM
002040 PHRFIL$="PHRSBLST" :PHRLIB$= INLIB$ :PHRVOL$= INVOL$ JM
002050 FER$=HEX(94)
002051 FOR J%=1 TO 3
002061 IF OPEN%(J%)=1 THEN CLOSE #J% :NEXT J%
002062 MAT OPEN% = ZER
002070 GOTO SCREEN1
002080 AGAN1: CLOSE #1 :CLOSE#2 :CLOSE#3
002090 GOTO GONE
002100 IO_ERR1: PRINT AT(23,1), "I/O ERROR ON COMMON WORD LIST FILE, #1":
002110 ;". FILE STATUS IS : ";FS(#1)
002120 STOP "PRESS ENTER TO CONTINUE"
002130 CLOSE #1 :CLOSE #2 :CLOSE #3 : GOTO SCREEN1
002140
002150 IO_ERR2: PRINT AT(23,1), "I/O ERROR ON WORD SUBST. LIST FILE, #2":
002160 ;". FILE STATUS IS : ";FS(#2)
002170 STOP "PRESS ENTER TO CONTINUE"

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 15:36 11/13/84 Page 5  
Input File is SCREEN in Library CRES@SRC on Volume CRES02

```
002180      CLOSE #1 : CLOSE #2 :CLOSE #3 :GOTO SCREEN1
002190
002200 IO_ERR3: PRINT AT(23,1), "I/O ERROR ON PHRASE SUBST. LIST FILE"; !
002210      " #3. FILE STATUS IS : ";FS(#3)
002220      STOP "PRESS ENTER TO CONTINUE"
002230      CLOSE #1 :CLOSE #2 :CLOSE #3 :GOTO SCREEN1
002240 GONE: IF RE%=16 OR RE%=32 THEN REG%=0 ELSE REG%=1
002250      END
```

Technical Report 86-001

THIS PAGE INTENTIONALLY LEFT BLANK.

### Document Analysis (ANALYSIS)

ANALYSIS in library CRES@SRC is the BASIC language program which performs the CRES analysis of a specified document. The object language version is ANALYSIS in library CRES@SAV. The analysis program is by far the most complex program in the CRES system, and it will be discussed in detail below.

At the start of the analysis program, it is necessary to initialize variables, set defaults, initialize the volumes and libraries and set up User File Blocks for each file to be used. Variables are initialized by specific assignments at the beginning of the program and by the internal subroutine INIT. Defaults are set by calling the external subroutine KIKOFF, which checks for the existence of a defaults file for the particular user. (See discussion of KIKOFF for details.) Volume and library variables are initialized using "EXTRACT", a Wang system utility. User File Blocks are established using BASIC "SELECT" commands.

The program next provides three screens, which are utilized to specify the document to be analyzed and the options to be employed in the analysis.

The first screen accepts the name of the document to be analyzed, and displays the default values for the Common Word List, the Word Substitute List and the Phrase Substitute List. Any of these defaults may be changed, and all file names are tested by the program to verify that they exist and are of the correct type.

The second screen accepts up to seven technical word lists to be used in the analysis, and these file names are verified to ensure data integrity.

The third screen presents all the possible options for a CRES analysis and presents default values. These defaults are the ones established for each user by the external subroutine KIKOFF the first time an analysis is run under that user ID. Any combination of the possible options may be selected. The options available to the user are:

- Report the Reading Grade Level of the document.
- Count the prepositions and flag sentences with "too many" prepositions. (If there are more than 2 prepositions in the sentence, and the ratio of total words in the sentence to prepositions is less than 5.4, the sentence is defined as containing "too many" prepositions.)

## Technical Report 86-001

- Flag passive voice. (A form of the verb "to be", followed by a past participle and immediately followed or preceded by the word "by", is defined as the full passive voice construction and flagged as such.)
- Flag long sentences. (If there are more than 22 words in the sentence it is considered a long sentence.)
- Flag long paragraphs. (If long paragraphs are to be flagged, the user has the option of specifying the maximum number of words and the maximum number of sentences to be allowed before the paragraph is flagged as long. The default values are 200 words or 10 sentences. The paragraph will be flagged after either maximum is encountered.)
- Flag double negatives.
- Flag and list uncommon words.
- Include substitutes. (If a substitute for a word or phrase appears on either the Word or Phrase Substitution List, the listed substitute(s) will be printed in brackets immediately following the given word or phrase.)
- Interim analyses. (If a lengthy text is to be analyzed, it may be desirable to perform interim analyses. A final, overall analysis will still be printed at the end of the report. The user can specify the number of pages between interim analyses. The default value is five pages.)
- Check personal sentences. (All sentences containing the words "you", "your", "I", "we", "my", or "our" are defined as personal sentences. At the end of the analysis printout, the total percentage of personal sentences contained in the text will be printed out.)
- Flag truncated passive voice. (Any form of the verb "to be", followed by a past participle, forms the truncated passive voice and will be flagged as such.)

In addition to these options, an automatic provision is made for flagging possible lists since they are considered highly undesirable in certain

technical writing applications. Whenever the phrase "consist(s) of" appears in a long sentence, followed by three or more commas, the query "LIST?" will be printed in the analysis.

After the user has entered the desired values on the third screen, tests are performed to ensure only logically correct selections have been made. Once all selections are ascertained to be valid, if a default file does not already exist, the program calls KIKOFF to store the current selections as the default values for that user. All technical word list names, as well as the analysis options specified, are inserted into this file. The default file name is formed by concatenating the user ID and the string "DFLTS", and it is stored in library CRES@DAT. The original defaults may be changed for any given analysis; however, the default file will remain unchanged and subsequent analyses will have the original defaults presented. To change the default file permanently, the user should scratch the existing default file through the "MANAGE FILES/LIBRARIES" capability of the Command Processor. Then, a new defaults file will be generated on the next analysis run using that user ID.

The analysis program must open each specified file. An output file, ANALYS##, is created and opened. The first analysis run will be ANALYS00, and subsequent runs will be numbered consecutively. A header is generated for this file which includes information on all files and options which have been selected. All variables for the analysis are initialized by the internal subroutine SETUP.

The text analysis proceeds word by word. The program uses the Wang VS word processing subroutines to access each word. If an End of File (EOF) is encountered, the current output line is ended and added to the output file. Final analysis results and an Uncommon Word List are also added, if those options were selected. If an EOF is not found, then the program calls the internal subroutine SPUNK. SPUNK strips leading and trailing punctuation or invalid characters from around the word. The internal subroutine SYLLY is then called to calculate the syllable count for this word. The syllables of the word are counted, using a number of both general and specific algorithms.

The syllable count is obtained by starting with the general rule that there is one syllable per vowel. Specific exceptions are made for the case of diphthongs or final "e"s. Then, provision is made for the exceptions to these cases. Certain syllables, or certain vowel and consonant patterns, have been found to be exceptions to these general rules, and separate provision is made for them. The words "area" and "areas" are exceptions to the evaluation algorithms but occur with sufficient frequency to make it worthwhile to treat them as a special case.

## Technical Report 86-001

The most efficient method of counting syllables was derived by combining a syllable count taken from the beginning of the word forward, together with one taken from the end of the word backward. Endings are stripped from the word, counted for the syllable count, and their letters removed from further consideration. Specific suffixes and prefixes are recognized and their syllables added to the count. A complete list of the special cases treated can be derived by consulting the code for the subroutine SYLLY in the ANALYSIS listing which follows this discussion.

For the purposes of the CRES analysis, certain words are defined as having only one syllable. The numbers one through nine are counted as one syllable each. Any word with a numeral appearing somewhere within it is also counted as having one syllable. Words which appear on a selected technical word list will have their Mandatory Technical Word flag checked. If the word is flagged as a Mandatory Technical Word, it is counted as a single syllable.

Certain information is obtained about the current word in the SPUNK and SYLLY subroutines of the ANALYSIS program. Appropriate flags are set as the information is obtained. If the word is found to contain invalid characters it is placed on the Uncommon Word List, if that option had been selected, and not processed any further. An invalid word will have no substitute listed in the Word Substitution List, and neither can it be part of a phrase included in the Phrase Substitution List. The invalid word, and any preceding words in the phrase buffer, can be added to the output line without further consideration.

If the word appears to be valid, each specified word list is searched for it until either the word is found or the lists are exhausted. If the word is not found on one of these lists, and if the option to flag and list uncommon words has been selected, then the word will be included on the Uncommon Word List. Uncommon words will not be analyzed any further. Any word which is found in the Acronym or Abbreviation Glossary is also exempted from further analysis.

If the word is found in one of the other lists, its attributes can be evaluated. Flags are set for later determination of preposition count, full or truncated passive voice, or double negatives, depending on the analysis options selected by the user. If the option to include substitutes has been selected, each word is considered as part of a possible phrase on the Phrase Substitution List. A phrase buffer holds the partial phrase as it is being constructed. As each word is added to the phrase buffer, the potential phrase is matched against the Phrase Substitution List. When a phrase is either completed or aborted by an ineligible word, the contents of the phrase buffer are added to an output buffer which holds the line of output text currently being built. If the option to include substitutes has not been selected, each

## Technical Report 86-001

word is immediately added to the output line being built. This output line is comprised of the original text and punctuation, with any of the editing suggestions and readability counts generated by the analysis.

After each page of output is generated, the user options are checked to determine whether an interim output should be printed. When the end of the document has been reached, if an interim analysis had been specified it is printed. Then the final reading grade level analysis and uncommon word list is generated. If no interim analyses were requested, only the final reading grade level analysis and uncommon word list are generated. In either case, normal indication of characteristics specified in the option selection such as uncommon words, passive voice, or long sentences will appear throughout the text.

Figure 4 presents a flowchart of the analysis program. A complete program listing with variable dictionary follows. A sample input text and the resultant CRES analysis output are included in appendix D.

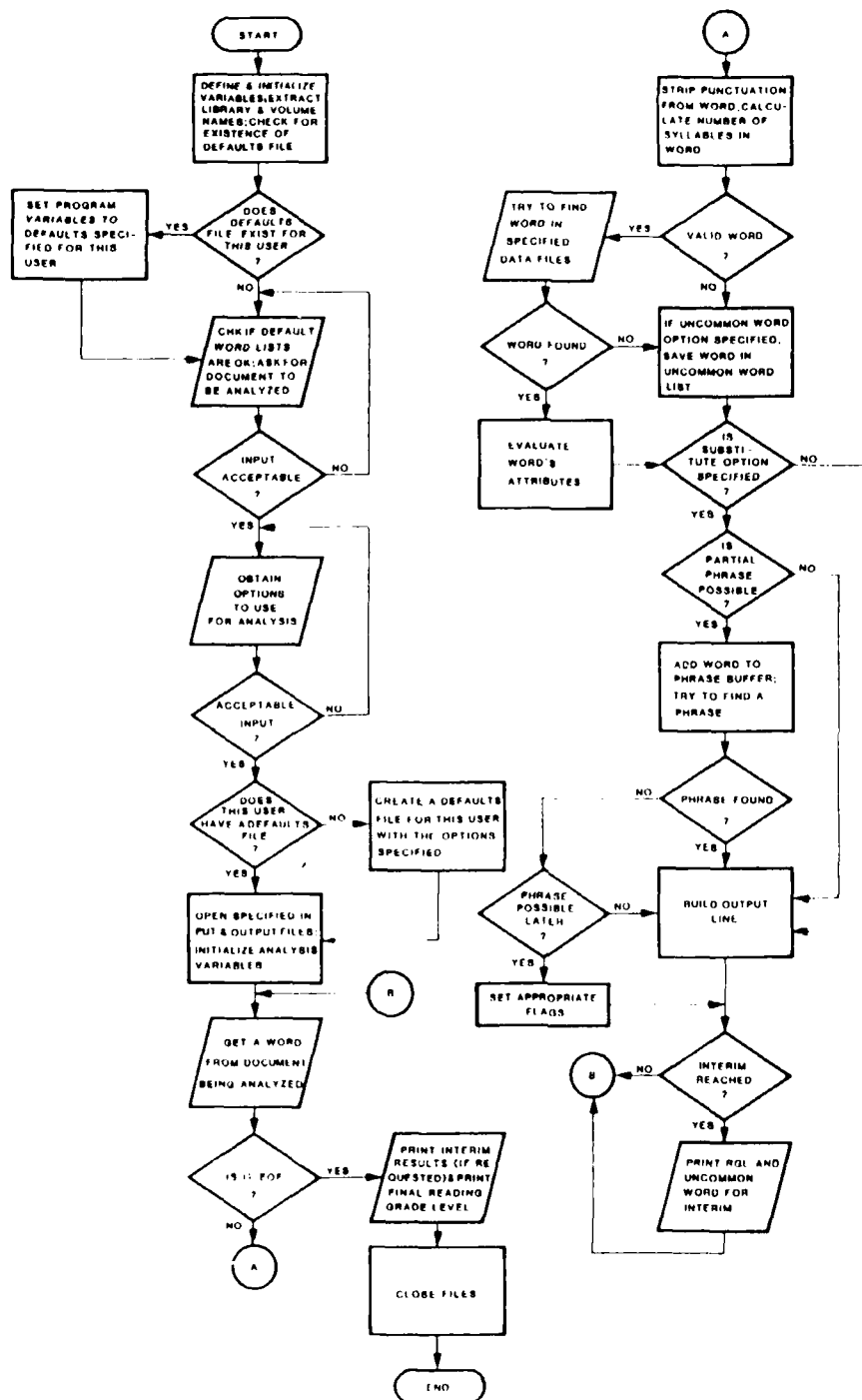


Figure 4. LRES Analysis Program Flowchart.

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 13:49 12/14/84 Page 1  
Input File is ANALYSIS in Library CRES@SRC on Volume CRES01

```

000010 /***** JM
000020 /* PROGRAM : ANALYSIS REVISED : 11/18/83 */ JM
000030 /* PROGRAMED BY: JOHN MEARS, LESLIE MONTGOMERY & SUSAN SCHALOW */ JM
000040 /* */ JM
000050 /* THIS IS THE ANALYSIS PROGRAM FOR THE VS CRES SYSTEM. */ JM
000060 /* THIS PROGRAM ALLOWS THE USER TO SPECIFY THE DOCUMENT */ JM
000070 /* TO BE ANALYZED, AND TO SPECIFY THE OPTIONS FOR THE */ JM
000080 /* ANALYSIS. ALL NECESSARY WORD LISTS AND FILES ARE */ JM
000090 /* OPENED, PREPATORY TO PERFORMING THE ANALYSIS. */ JM
000100 /* */ JM
000110 /* NOTE: WPSBRTNB IN LIBRARY @LINKLIB, KICKOFF IN LIBRARY */ JM
000120 /* ----- */ JM
000130 /* CRES@SAV, SCREEN IN LIBRARY CRES@SAV, AND TITLPAGE */ JM
000140 /* ----- */ JM
000150 /* IN LIBRARY CRES@SAV ARE LINKED THROUGH THE FILES OPTION. */ JM
000160 /* ----- */ JM
000170 /* WPSBRTNB HANDLES ALL WORD PROCESSING COMMANDS. */ JM
000180 /* KICKOFF HANDLES THE DEFAULTS FILE FOR USER DEFAULTS. */ JM
000190 /* SCREEN HANDLES THE FIRST SCREEN ASKING FOR FILENAMES. */ JM
000200 /* TITLPAGE HANDLES THE HEADER FOR THE OUTPUT FILE. */ JM
000210 /* */ JM
000220 /***** JM
000230 DIM DOCFIL$8, DOCLIB$8, DOCVOL$6, TECFIL$(7)8, TECLIB$(7)8 JM
000240 DIM TECVOL$(7)6, PAG_TITL$60, OPEN$(13), RC$1, MSG$40, LIB$1 JM
000250 DIM RGL$1, PREP_CNT$1, PASSV$1, LNG_PAR$1, IOBUF$190, L$1 JM
000260 DIM UNCOM$1, NEG$1, INT_ANL$1, NUM_SUB$1, FC$(10)1 JM
000270 DIM ADVFN$1, LNG_CLS$1, PERS_SENT$1, LNG_SEN$1, ABBRE$(500)20 JM
000280 DIM ID$5, PRINTFL$8, PLIB$8, PVOL$6, TABLE$(256)1 JM
000290 DIM WORDP$24, TITLBUF$77, UCW$(500)20, UCWCT$(500), C$2, LW$(5) JM
000300 DIM UCWFLG$(5)1, LSENT$(5), PREPFL$(5), PASSVF$(5)1, DNFL$(5)1 JM
000310 DIM NOUNFL$(5)1, LCLSFL$(5), SAVPH$(5)20, SAVPHL$(5), WSL$1 JM
000320 DIM LAST$1, NP$2, TRNK_PV$1, SAVSYL$(5), LNGWDS$3, LNGSNT$2 JM
000330 DIM NUM_SUBZ$1, ENTER$5, PF$4, PF1$3, PF15$4, Q$1, RUNLIB$8, RUNVOL$6 JM
000340 DIM ER$79, MSG2$40, MSK$2, CMSC$255, WORDQ$24, BUFFA$(40)77, RE$1 JM
000350 DIM UNC$(20)20, F1$4, E$65, FX$1, IOBUFF$40, INLIB$8, INVOL$6 JM
000360 /* */ JM
000370 /***** JM
000380 /* */ JM
000390 /* VARIABLE DICTIONARY FOR PROGRAM ANALYSIS: */ JM
000400 /* ===== */ JM
000410 /* */ JM
000420 /* ABBRE$ : ARRAY TO HOLD COMMONLY USED ABBREVIATIONS AND */ JM
000430 /* MNEMONICS; */ JM
000440 /* ABBRE% : LENGTH OF ABBRE$ ARRAY; */ JM
000450 /* ADVFN$ : A FLAG TO TELL IF ADVANCED FUNCTIONS ARE REQUESTED; */ JM
000460 /* BEGINY$ : USED TO COMPUTE STARTING PAGE NUMBER FOR PRTRGL; */ JM
000470 /* BLANK$ : A FLAG USED TO CATCH PHRASES WITHOUT SUBSTITUTES; */ JM
000480 /* BP% : BREAK POINT; */ JM
000490 /* BUFFA$ : BUFFER TO PASS BACK INFO FOR PRINTFILE HEADER; */ JM
000500 /* BUFFERS$ : BUFFER TO HOLD MESSAGE FOR INTERIM OR FINAL RGL */ JM
000510 /* REPORT; */ JM
000520 /* C$ : CONTROL CHARACTER FOR FORMAT STATEMENTS; */ JM
000530 /* CLSPOS$ : FLAG TO TELL IF END/START OF A CLAUSE; HAS VALUES */ JM
000540 /* 'B'=BEGINNING, 'E'=END, 'Y'=INTERMEDIATE; */ JM
000550 /* CW$ : COMMON WORD FLAG; HAS VALUES ' '=ALLRIGHT(COMMON WD.), */ JM
000560 /* 'Y'=PHRASE FOUND, 'N'=UNCOMMON WORD, 'X'=RETRY */ JM
000570 /* D% : FLAG TO TELL US RETURN CODE FROM PRT_WDS */ JM

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 13:49 12/14/84 Page 2  
Input File is ANALYSIS in Library CRES@SRC on Volume CRES01

```

000580 /* DNFL$ : AN ARRAY TO MARK THE LOCATION OF A DOUBLE NEGATIVE */ JM
000590 /* OCCURRENCE; */ JM
000600 /* DOCFIL$ : THE NAME OF THE INPUT DOCUMENT FILE; */ JM
000610 /* DOCLIB$ : THE NAME OF THE INPUT DOCUMENT LIBRARY; */ JM
000620 /* DOCVOL$ : THE NAME OF THE INPUT DOCUMENT VOLUME; */ JM
000630 /* ENDY$ : USED TO COMPUTE ENDING PAGE NUMBER FOR PRTRGL; */ JM
000640 /* END_O_PAR$ : END OF PARAGRAPH; */ JM
000650 /* E_O_F$ : FLAG TO INDICATE END OF FILE; */ JM
000660 /* FC$ : AN ARRAY OF FAC'S USED IN WORD LISTS SELECTION; */ JM
000670 /* FLAG$ : A FLAG TO INDICATE TROUBLE IN FILE SELECTION IF */ JM
000680 /* FLAG=1; */ JM
000690 /* POUND$ : WORD IN WORD LIST FLAG; HAS VALUES ' '=CLEARING */ JM
000700 /* FLAG, 'Y'=WORD FOUND; */ JM
000710 /* F1 : PERCENT OF UNCOMMON WORDS; */ JM
000720 /* F2 : PERCENT OF LONG SENTENCES */ JM
000730 /* HOME$ : VARIABLE USED TO DETERMINE IF USER WISHES TO END */ JM
000740 /* ANALYSIS NOW; HOME=0 MEANS NO, =1 MEANS YES; */ JM
000750 /* ID$ : USED AS A PARAMETER FOR 'WOPEN' STATEMENT; CREATED */ JM
000760 /* BY CONCATONATING DOCUMENT # WITH VOLUME LETTER; */ JM
000770 /* IDX$ : POINTER IN UNCOMMON WORD TABLE; */ JM
000780 /* IL$ : POINTER TO ILINE$ TO SHOW POSITION IN INPUT LINE; */ JM
000790 /* ILINE$ : BUFFER TO HOLD EACH LINE OF INPUT; */ JM
000800 /* INT_ANL$ : INTERIM ANALYSIS OPTION; TELLS US TO PRINT */ JM
000810 /* SUMMARY READING GRADE LEVEL; */ JM
000820 /* INVLD$ : INVALID WORD FLAG; HAS VALUES ' ','Y'=INVALID WORD, */ JM
000830 /* 'P'=NUMBER OR GARBAGE */ JM
000840 /* IO$ : POSITION IN OUTPUT LINE (OLINE$); */ JM
000850 /* IOBUF$ : BUFFER USED TO READ INFO FROM THE ABBREVIATION/ */ JM
000860 /* MNEMONIC LIST; */ JM
000862 /* IP$ : POINTER TO THE PHRASE BUFFER; */ JM
000880 /* IW$ : A POINTER TO THE ARRAY OF WORDS LAST FOUND (WORDS$); */ JM
000890 /* L$ : VARIABLE USED TO CATCH TRAILING PUNCTUATION AND PLACE */ JM
000900 /* IT OUTSIDE OF THE SQUARE BRACKETS(' [ ]') WHEN FLAGGING */ JM
000910 /* UNCOMMON WORDS; */ JM
000920 /* L1$,L2$ : VARIABLES USED TO EXTRACT SUBSTITUTES FROM WORD */ JM
000930 /* SUBSTITUTION LIST; */ JM
000940 /* LAST$ : FLAG TO TELL IF INTERIM RGL HEADER NEEDED OR TOTAL */ JM
000950 /* RGL HEADER NEEDED; */ JM
000960 /* LC$ : THE CURRENT LINE COUNT; */ JM
000970 /* LEN$ : WORD LENGTH; */ JM
000980 /* LIB$ : THE 1 CHARACTER WORD PROCESSING DOCUMENT LIBRARY */ JM
000990 /* INPUT BY THE USER; USED TO CREATE THE 8 CHARACTER */ JM
001000 /* DOCLIB$ VARIABLE; */ JM
001010 /* LIST$ : A COUNTER OF COMMAS IN LONG SENTENCES FOR 'LIST?' */ JM
001020 /* REMARK IN LONG SENTENCES; */ JM
001030 /* LL$ : LINE LENGTH; */ JM
001040 /* LNGSNT$ : THE NUMBER OF SENTENCES TO FLAG A LONG SENTENCE; */ JM
001050 /* LNGWDS$ : THE NUMBER OF WORDS TO FLAG A LONG SENTENCE; */ JM
001060 /* LNG_CLS$ : LONG CLAUSE OPTION; */ JM
001070 /* LNG_PAR$ : LONG PARAGRAPH OPTION; */ JM
001080 /* LNG_SEN$ : LONG SENTENCE OPTION; */ JM
001090 /* LP$ : LINES PER PAGE; SET AT 50; */ JM
001100 /* LS$ : NUMBER OF LONG SENTENCES; */ JM
001110 /* LSENT$() : AN ARRAY USED TO MARK THE COUNT AND WHERE THE */ JM
001120 /* END OF A LONG SENTENCE FALLS; */ JM
001130 /* LW$() : ARRAY TO HOLD THE LENGTH OF EACH WORD IN WORDS$; */ JM
001140 /* MSG$ : THE MESSAGE RETURNED BY THE "LIMITS" SUBROUTINE; */ JM

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 13:49 12/14/84 Page 3  
Input File is ANALYSIS in Library CRES@SRC on Volume CRES01

```

001150 /* NC% : NUMBER OF WORDS PER CLAUSE; */ JM
001160 /* NEG$ : NEGATIVE OPTION FLAG; */ JM
001170 /* NEG(1) : IF = 2 THEN DOUBLE NEGATIVE COUNT; */ JM
001180 /* NEG(2) : DITTO; */ JM
001190 /* NEG(3) : NEGATIVE FLAG; */ JM
001200 /* NP$ : (CHARACTER) PAGES TO ANALYZE BEFORE REPORTING RGL; */ JM
001210 /* NP% : (INTEGER) PAGES TO ANALYZE BEFORE REPORTING RGL; */ JM
001220 /* NSW : NUMBER OF SYLLABLES PER WORD; */ JM
001230 /* NUM_SUB$ : NUMBER OF SUBSTITUTES OPTION; INITIALIZED TO 2; */ JM
001240 /* NWS : NUMBER OF WORDS PER SENTENCE; */ JM
001250 /* OLINE$ : THE OUTPUT LINE BUFFER; */ JM
001260 /* OPEN% : AN ARRAY OF FLAGS TO INDICATE WHICH FILES ARE OPEN; */ JM
001270 /* PAG_TITL$ : PAGE TITLE INPUT BY USER; */ JM
001280 /* PASSV$ : REGULAR PASSIVE VOICE OPTION FLAG; */ JM
001290 /* PASSV% : PASSIVE FLAG; 0=CLEAR, 7=FLAG SET; */ JM
001300 /* PASSVF$ : AN ARRAY TO MARK THE LOCATION (IN A SENTENCE) OF */ JM
001310 /* PASSIVE OR TRUNCATED PASSIVE VOICE; */ JM
001320 /* PC% : PAGE COUNT; */ JM
001330 /* PE% : PERSONAL SENTENCE COUNT; */ JM
001340 /* PERSENT$ : PERSONAL SENTENCES; */ JM
001350 /* PERS_SENT$ : PERSONAL SENTENCE OPTION; */ JM
001360 /* PFLAG% : FLAG TO ASSIST WITH PRTLNE SUBROUTINE; TELLS US */ JM
001370 /* THE STATUS AFTER RETURN; */ JM
001380 /* PHCT% : A COUNT OF SAVED WORDS FROM POSSIBLE PHRASES; */ JM
001390 /* PLIB$ : THE LIBRARY THAT THE OUTPUT FILE GOES IN; */ JM
001400 /* PREPFL%() : AN ARRAY USED TO MARK IF THERE ARE TOO MANY */ JM
001410 /* PREPOSITIONS, THE AMOUNT OF PREPOSITIONS, AND */ JM
001420 /* THE LOCATION IN THE SENTENCE OF THE MARKING; */ JM
001430 /* PREP_CNT$ : PREPOSITION COUNT OPTION; */ JM
001440 /* PRINTFL$ : THE FILENAME OF THE OUTPUT PRINT FILE; */ JM
001450 /* PS% : PREPOSITION COUNT FOR EACH SENTENCE; */ JM
001460 /* PVOL$ : THE VOLUME THAT THE OUTPUT FILE GOES IN; */ JM
001470 /* RC$ : THE RETURN CODE FROM THE "LIMITS" SYSTEM SUBROUTINE; */ JM
001480 /* RECORD$ : THE RECORD FOUND ON A WORD LIST OF THE SPECIFIC */ JM
001490 /* WORD WE'RE LOOKING FOR; */ JM
001500 /* RECL$ : THE RECORD FOUND CONTAINING THE SPECIFIC WORD AND */ JM
001510 /* UP TO TWO SUBSTITUTES; */ JM
001520 /* RGL$ : READING GRADE LEVEL OPTION; */ JM
001530 /* SAVPH$ : AN ARRAY USED TO SAVE WORDS OF POSSIBLE PHRASES; */ JM
001540 /* SAVPHL% : AN ARRAY USED TO SAVE THE LENGTH(S) OF WORDS OF */ JM
001550 /* POSSIBLE PHRASES; */ JM
001560 /* SAVSYL% : AN ARRAY USED TO SAVE THE SYLLABLE COUNT(S) OF */ JM
001570 /* WORDS OF POSSIBLE PHRASES; */ JM
001580 /* SC% : SENTENCE COUNT; */ JM
001590 /* SENTPOSS$ : END/START OF A SENTENCE; VALUES ARE 'B'=BEGIN, */ JM
001600 /* 'E'=ENDING; */ JM
001610 /* SLC% : SYLLABLE COUNT SUBTOTAL FOR THIS PASSAGE; */ JM
001620 /* SP% : SENTENCES PER PARAGRAPH COUNT; */ JM
001630 /* TABLE$ : AN ARRAY TO HOLD ALL CONTROL CHARACTERS WE SHOULD */ JM
001640 /* NOT BE CONCERNED WITH; */ JM
001650 /* TECFIL$ : AN ARRAY TO HOLD THE FILE NAMES FOR THE TECHNICAL */ JM
001660 /* WORD LISTS INPUT; */ JM
001670 /* TECH% : FLAG TO TELL IF WORD IS PRESENT ON TECHNICAL WORD */ JM
001680 /* LISTS; IF SO, TECH=1; ELSE TECH=0; */ JM
001690 /* TECLIB$ : AN ARRAY TO HOLD THE LIBRARY NAMES FOR THE */ JM
001700 /* TECHNICAL WORD LISTS INPUT; */ JM
001710 /* TECVOL$ : AN ARRAY TO HOLD THE VOLUME NAMES FOR THE */ JM

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 13:49 12/14/84 Page 4  
Input File is ANALYSIS in Library CRES@SRC on Volume CRES01

```

001720 /*          TECHNICAL WORD LISTS INPUT;                */ JM
001730 /*  TITLBUF$ : BUFFER TO HOLD CENTERED TITLE FOR OUTPUT; */ JM
001740 /*  TLC% : SYLLABLE COUNT FOR EACH WORD;                */ JM
001750 /*  TPASSV% : FLAG TO KEEP TRACK OF TRUNCATED PASSIVE VOICE; */ JM
001760 /*          TPASSV%=1 IMPLIES 'TO BE' FOUND,=3 IMPLIES 'TO BE' */ JM
001770 /*          + 'PAST PARTICIPLE' FOUND;                    */ JM
001780 /*  TRNK_PV$ : TRUNCATED PASSIVE VOICE OPTION; "Y" OR "N"; */ JM
001790 /*  UCW$ : A LARGE ARRAY TO HOLD UNCOMMON WORDS;          */ JM
001800 /*  UCWCT% : A LARGE NUMERIC ARRAY TO HOLD FREQUENCIES OF EACH */ JM
001810 /*          ENTRY IN THE UCW$;                             */ JM
001820 /*  UCWFLG$ : AN ARRAY USED TO TELL IF FORMER POSSIBLE PHRASE */ JM
001830 /*          WORDS ARE UNCOMMON;                             */ JM
001840 /*  UNCS$ : A BUFFER TO HOLD ALL UNCOMMON WORDS FOR EACH */ JM
001850 /*          SENTENCE; WHEN END OF SENTENCE FOUND, ADDED TO UCW$; */ JM
001860 /*  UNCPT% : A POINTER TO THE NEXT AVAILABLE LOCATION IN THE */ JM
001870 /*          UNCS$ ARRAY BUFFER FOR UNCOMMON WORDS;          */ JM
001880 /*  UNCOM$ : UNCOMMON WORD OPTION; IF = 'Y', THEN SAVE UNCOMMON */ JM
001890 /*          WORDS AND LIST AFTER EACH INTERIM ANALYSIS;      */ JM
001900 /*  UPDMSG$ : THE MESSAGE FLASHED TO THE USER AFTER INCREMENTS */ JM
001910 /*          OF 250 WORDS HAVE BEEN EXAMINED IN UPDATE SCREEN; */ JM
001920 /*  UW% : NUMBER OF UNCOMMON WORDS SO FAR;                  */ JM
001930 /*  WC% : WORD COUNT;                                        */ JM
001940 /*  WORDP$ : THE BUFFER USED TO BUILD POSSIBLE PHRASES;      */ JM
001950 /*  WORDS$() : A SINGLE DIMENSION ARRAY TO HOLD WORDS AND THEIR */ JM
001960 /*          SUBSTITUTES;                                      */ JM
001970 /*  WORD1$ : THE INPUT WORD IN ALL CAPITALS;                 */ JM
001980 /*  WORD2$ : HOLDS THE INPUT WORD WHILE SYLLY DEALS WITH EACH */ JM
001990 /*          PART OF A WORD SEPERATED BY A SLASH ("/") OR A */ JM
002000 /*          HYPHEN ("-");                                     */ JM
002010 /*  WS% : WORDS PER SENTENCE COUNT;                          */ JM
002020 /*  WSL$ : FLAG TO TELL IF WORD FOUND IN WORD SUBSTITUTION LIST; */ JM
002030 /*          "N" = NO & "Y" = YES;                            */ JM
002040 /*  WW1% : USED TO COUNT THE NUMBER OF WORDS PROCESSED FOR USER */ JM
002050 /*          UPDATE OF PROGRESS;                                */ JM
002060 /*  XLS% : THE GRAND TOTAL OF LONG SENTENCES FOR ALL ANALYSES; */ JM
002070 /*  XSC% : THE GRAND TOTAL OF SENTENCES FOR ALL ANALYSES;     */ JM
002080 /*  XSLC% : THE GRAND TOTAL OF SYLLABLES FOR ALL ANALYSES;    */ JM
002090 /*  XUC% : THE GRAND TOTAL OF UNCOMMON WORDS FOR ALL ANALYSES; */ JM
002100 /*  XWC% : THE GRAND TOTAL OF WORDS FOR ANALYSES;             */ JM
002110 /*          */ JM
002120 /****** */ JM
002130  F1$=HEX(A01400AA) /* FORMAT FOR READING ABBREVIATIONS */ JM
002140  RE$, RGL$, PREP_CNT$, PASSV$, LNG_PAR$ = "Y" JM
002150  LNG_SEN$, PERS_SENT$, UNCOM$, NEG$, INT_ANL$ = "Y" JM
002160  NUM_SUB2$, NUM_SUB$, TRNK_PV$, LNG_CLS$ = "Y" JM
002170  WSL$,ADVFNS$="N" JM
002180  NP%=5 JM
002190  HOME%=0 JM
002200  ENTER$="ENTER" JM
002210  PF$="PF16" JM
002220  PF15$="PF15" JM
002230  LNGWDS$="200" JM
002240  LNGSNT$="10" JM
002250  NP$="05" JM
002260  TECFIL$(), TECLIB$(), TECVOL$() = ALL(HEX(20)) JM
002270 JM
002280  CALL "EXTRACT" ADDR("IV",INVOL$,"SV",PVOL$,"IL",INLIB$,"SL", : JM

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 13:49 12/14/84 Page 5  
Input File is ANALYSIS in Library CRES@SRC on Volume CRES01

```

002290      PLIB$, "RL", RUNLIB$, "RV", RUNVOL$)                JM
002300      SELECT #1, "WPDOCMT", VARC, INDEXED, RECSIZE=256,    ! JM
002310      KEYPOS=01, KEYLEN=20, IOERR GOTO IO_ERR1            JM
002320      SELECT #2, "CMNWDLST", VARC, INDEXED, RECSIZE=40,    ! JM
002330      KEYPOS=01, KEYLEN=20, IOERR GOTO IO_ERR2            JM
002340      SELECT #3, "WRDSBLST", VARC, INDEXED, RECSIZE=80,    ! JM
002350      KEYPOS=01, KEYLEN=20, IOERR GOTO IO_ERR3            JM
002360      SELECT #4, "PHRSBLST", VARC, INDEXED, RECSIZE=70,    ! JM
002370      KEYPOS=01, KEYLEN=20, IOERR GOTO IO_ERR4            JM
002380      SELECT #5, "TECHWL1", VARC, INDEXED, RECSIZE=40,    ! JM
002390      KEYPOS=01, KEYLEN=20, IOERR GOTO IO_ERR5            JM
002400      SELECT #6, "TECHWL2", VARC, INDEXED, RECSIZE=40,    ! JM
002410      KEYPOS=01, KEYLEN=20, IOERR GOTO IO_ERR6            JM
002420      SELECT #7, "TECHWL3", VARC, INDEXED, RECSIZE=40,    ! JM
002430      KEYPOS=01, KEYLEN=20, IOERR GOTO IO_ERR7            JM
002440      SELECT #8, "TECHWL4", VARC, INDEXED, RECSIZE=40,    ! JM
002450      KEYPOS=01, KEYLEN=20, IOERR GOTO IO_ERR10           JM
002460      SELECT #9, "TECHWL5", VARC, INDEXED, RECSIZE=40,    ! JM
002470      KEYPOS=01, KEYLEN=20, IOERR GOTO IO_ERR11          JM
002480      SELECT #10, "TECHWL6", VARC, INDEXED, RECSIZE=40,   ! JM
002490      KEYPOS=01, KEYLEN=20, IOERR GOTO IO_ERR12           JM
002500      SELECT #11, "TECHWL7", VARC, INDEXED, RECSIZE=40,   ! JM
002510      KEYPOS=01, KEYLEN=20, IOERR GOTO IO_ERR13           JM
002520      SELECT #12, "PRNTFILE", PRINTER, RECSIZE=80, IOERR GOTO IO_ERR8 JM
002530      SELECT #13, "ABBREFIL", VARC, INDEXED, RECSIZE=190, ! JM
002540      KEYPOS=01, KEYLEN=20, IOERR GOTO IO_ERR14           JM
002550      SELECT #14, "TABLE", CONSEC, RECSIZE=256, IOERR GOTO IO_ERR9 JM
002560      CALL "KIKOFF" ADDR(RGL$, PREP CNT$, PASSV$, LNG SEN$, LNG PAR$, ! JM
002570      LNGWDS$, LNGSNT$, NEG$, UNCOM$, NUM SUB$, INT ANL$, NP$, ADVFN$, ! JM
002580      PERS SENT$, LNG CLS$, TRNK PV$, TECFIL$(), TECLIB$(), TECVOL$(), RE$, ! JM
002581      INLIB$, INVOL$)                                     JM
002590      DOCFIL$=" "                                         JM
002600      DOCVOL$, COMVOL$, PHRVOL$, WRDVOL$=INVOL$          JM
002610      COMLIB$, PHRLIB$, WRDLIB$=INLIB$                  JM
002620      SCRNL:                                             JM
002630      CALL "SCRNL" (COMFIL$, COMLIB$, COMVOL$, WRDFIL$, WRDLIB$, ! JM
002640      WRDVOL$, PHRFIL$, PHRLIB$, PHRVOL$, DOCFIL$, DOCLIB$, DOCVOL$, ! JM
002650      PAG TITL$, LIB$, REG$, INLIB$, INVOL$)              JM
002660      IF REG%=0 THEN ENDPRG                                JM
002670      FC$()=ALL(HEX(81)) : FC$(8)=HEX(82) : FC$(9)=HEX(80) JM
002680      FX$=HEX(8C) : E$=" "                                JM
002690                                                         JM
002700      /* GET NAME OF FILES TO USE IN PROG. AND PAGE HEADER */ JM
002710                                                         JM
002720      SCRNL: ACCEPT                                       ! JM
002730      AT (01,03),                                         ! JM
002740      "CRES: Document Analysis Subsystem (Technical File Sele: JM
002750      ction Screen)",                                     ! JM
002760      AT (02,03),                                         ! JM
002770      "=====": JM
002780      "=====": JM
002790      AT (04,10),                                         ! JM
002800      "Please identify up to 7 Technical Word Lists to be used in", ! JM
002810      AT (05,10),                                         ! JM
002820      "the analysis.",                                     ! JM
002830      AT (07,10),                                         ! JM
002840      PAC(FX$), E$,                                     ! JM

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 13:49 12/14/84 Page 6  
Input File is ANALYSIS in Library CRES@SRC on Volume CRES01

```

002850      AT (09,09),                                : JM
002860 "1. FILE :",                                      : JM
002870      AT (09,20), FAC(FC$(01)), TECFIL$(1)         , CH(08), : JM
002880      AT (09,32),                                      : JM
002890 "LIBRARY :",                                      : JM
002900      AT (09,43), FAC(FC$(01)), TECLIB$(1)          , CH(08), : JM
002910      AT (09,55),                                      : JM
002920 "VOLUME :",                                      : JM
002930      AT (09,65), FAC(FC$(01)), TECVOL$(1)         , CH(06), : JM
002940      AT (10,09),                                      : JM
002950 "2. FILE :",                                      : JM
002960      AT (10,20), FAC(FC$(02)), TECFIL$(2)         , CH(08), : JM
002970      AT (10,32),                                      : JM
002980 "LIBRARY :",                                      : JM
002990      AT (10,43), FAC(FC$(02)), TECLIB$(2)          , CH(08), : JM
003000      AT (10,55),                                      : JM
003010 "VOLUME :",                                      : JM
003020      AT (10,65), FAC(FC$(02)), TECVOL$(2)         , CH(06), : JM
003030      AT (11,09),                                      : JM
003040 "3. FILE :",                                      : JM
003050      AT (11,20), FAC(FC$(03)), TECFIL$(3)         , CH(08), : JM
003060      AT (11,32),                                      : JM
003070 "LIBRARY :",                                      : JM
003080      AT (11,43), FAC(FC$(03)), TECLIB$(3)          , CH(08), : JM
003090      AT (11,55),                                      : JM
003100 "VOLUME :",                                      : JM
003110      AT (11,65), FAC(FC$(03)), TECVOL$(3)         , CH(06), : JM
003120      AT (12,09),                                      : JM
003130 "4. FILE :",                                      : JM
003140      AT (12,20), FAC(FC$(04)), TECFIL$(4)         , CH(08), : JM
003150      AT (12,32),                                      : JM
003160 "LIBRARY :",                                      : JM
003170      AT (12,43), FAC(FC$(04)), TECLIB$(4)          , CH(08), : JM
003180      AT (12,55),                                      : JM
003190 "VOLUME :",                                      : JM
003200      AT (12,65), FAC(FC$(04)), TECVOL$(4)         , CH(06), : JM
003210      AT (13,09),                                      : JM
003220 "5. FILE :",                                      : JM
003230      AT (13,20), FAC(FC$(05)), TECFIL$(5)         , CH(08), : JM
003240      AT (13,32),                                      : JM
003250 "LIBRARY :",                                      : JM
003260      AT (13,43), FAC(FC$(05)), TECLIB$(5)          , CH(08), : JM
003270      AT (13,55),                                      : JM
003280 "VOLUME :",                                      : JM
003290      AT (13,65), FAC(FC$(05)), TECVOL$(5)         , CH(06), : JM
003300      AT (14,09),                                      : JM
003310 "6. FILE :",                                      : JM
003320      AT (14,20), FAC(FC$(06)), TECFIL$(6)         , CH(08), : JM
003330      AT (14,32),                                      : JM
003340 "LIBRARY :",                                      : JM
003350      AT (14,43), FAC(FC$(06)), TECLIB$(6)          , CH(08), : JM
003360      AT (14,55),                                      : JM
003370 "VOLUME :",                                      : JM
003380      AT (14,65), FAC(FC$(06)), TECVOL$(6)         , CH(06), : JM
003390      AT (15,09),                                      : JM
003400 "7. FILE :",                                      : JM
003410      AT (15,20), FAC(FC$(07)), TECFIL$(7)         , CH(08), : JM

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 13:49 12/14/84 Page 7  
Input File is ANALYSIS in Library CRES@SRC on Volume CRES01

```

003420      AT (15,32),                      : JM
003430 "LIBRARY :",                          : JM
003440      AT (15,43), FAC(FC$(07)), TECLIB$(7) , CH(08), : JM
003450      AT (15,55),                          : JM
003460 "VOLUME :",                            : JM
003470      AT (15,65), FAC(FC$(07)), TECVOL$(7) , CH(06), : JM
003480      AT (21,28),                          : JM
003490 "Press",                               : JM
003500      AT (21,34), FAC(HEX(84)), ENTER$,      : JM
003510      AT (21,40),                          : JM
003520 "to continue ",                       : JM
003530      AT (22,19),                          : JM
003540 "Press",                               : JM
003550      AT (22,25), FAC(HEX(84)), PF15$,      : JM
003560      AT (22,30),                          : JM
003570 "to return to the previous menu",        : JM
003580      AT (23,18),                          : JM
003590 "Press",                               : JM
003600      AT (23,24), FAC(HEX(84)), PF$,        : JM
003610      AT (23,29),                          : JM
003620 "to return to the CRES Master Menu",      : JM
003630 KEYS(HEX(0010200F1F)), KEY(REC%)         JM
003640      E$=" "                               JM
003650      IF REC%=16 OR REC%=32 THEN ENDPRG      JM
003660      IF REC%=15 OR REC%=31 THEN SCRN        JM
003670      XWC%,XSC%,XSLC%,XUC%,XLS%=0          :LAST$="N" JM
003680      FLAG% = 0 : DOCLIB$=" "              :WW1%=0 JM
003690      MAT LSENT% = ZER /* INITIALIZE VARIABLES FOR ANALYSIS */ JM
003700      IF NUM_SUB$="0" THEN NUM_SUB$="N"      JM
003710      IF NUM_SUB$="2" THEN NUM_SUB$="Y"      JM
003720                                           JM
003730 /* SEE IF FILES EXIST THAT THE USER INPUT */ JM
003740                                           JM
003750      FC$( )=ALL(HEX(81)) : FC$(8)=HEX(82) : FC$(9)=HEX(80) JM
003760                                           JM
003770      FOR I%=1 TO 7                          JM
003780          FC$(I%)=HEX(81)                     JM
003790          IF TECFIL$(I%)=" " AND TECLIB$(I%)=" " AND : JM
003800              TECVOL$(I%)=" " THEN NI          JM
003810          CALL "LIMITS"(TECFIL$(I%),TECLIB$(I%),TECVOL$(I%),RC$,MSG$) JM
003820          IF RC$="Y" THEN CALL "READFDR" ADDR(TECFIL$(I%), : JM
003830              TECLIB$(I%),TECVOL$(I%),0%,"RS",RAG%,RET%) JM
003840          IF RC$<>"Y" OR RAG%<>40 THEN MESS      JM
003850          J%=I%+4                              JM
003860          OPEN NOGETPARM #J%, SHARED, FILE=TECFIL$(I%), : JM
003870              LIBRARY=TECLIB$(I%), VOLUME=TECVOL$(I%) JM
003880          READ #J%, IOBUFF$                     JM
003890          CLOSE #J%                             JM
003900          IF STR(IOBUFF$,1,16)=" TECHNICAL WORDS" THEN NI JM
003910      MESS: FC$(I%)=HEX(91)                     JM
003920      IF RC$<>"Y" THEN E$=" Indicated file does not exist." : JM
003930      ELSE E$=" Indicated file is NOT a Technical Word List." JM
003940          FX$=HEX(84)                             JM
003950          I%=7                                       JM
003960          FLAG% = 1                                 JM
003970      NI: NEXT I%                                 JM
003980

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 13:49 12/14/84 Page 8  
Input File is ANALYSIS in Library CRES@SRC on Volume CRES01

```

003990 IF FLAG=1 THEN SCREEN1 JM
004000 FOR I=1 TO 6 /* MAKE SURE USER DIDN'T SPECIFY DUPLICATES */ JM
004010 FOR JI=I+1 TO 7 JM
004020 IF TECFIL$(I)=" " THEN JI=8 JM
004030 IF TECFIL$(I)=" " THEN NIN JM
004040 IF TECFIL$(I)<>TECFIL$(JI) THEN NIN JM
004050 IF TECLIB$(I)<>TECLIB$(JI) THEN NIN JM
004060 IF TECVOL$(I)<>TECVOL$(JI) THEN NIN /* DIFF. VOLS */ JM
004070 FC$(I)=HEX(91) /* SET THE FAC'S */ JM
004080 FC$(JI)=HEX(91) JM
004090 ES=" Duplicate Lists were specified." JM
004100 FX$=HEX(84) JM
004110 I=7 /* GET OUT OF THE LOOP */ JM
004120 JI=8 JM
004130 FLAG=1 /* SET THE FLAG */ JM
004140 NIN: NEXT JI JM
004150 NEXT I JM
004160 IF FLAG=1 THEN SCREEN1 JM
004170 IF PAG_TITL$=" " THEN OPT_SCRN JM
004180 IDENT$=((LEN(PAG_TITL$))/2) JM
004190 TITLBUF$=ALL(HEX(20)) JM
004200 STR(TITLBUF$,34-IDENT$)=PAG_TITL$ JM
004210 JM
004220 /* GET OPTIONS FROM THE USER. */ JM
004230 JM
004240 OPT_SCRN: JM
004250 IF NUM_SUB$="2" THEN NUM_SUB$="Y" JM
004260 IF NUM_SUB$="0" THEN NUM_SUB$="N" JM
004270 ACCEPT : JM
004280 AT (01,03), : JM
004290 "CRES: Document Analysis Subsystem (Option Sele: JM
004300 ction Screen)", : JM
004310 AT (02,03), : JM
004320 "===== : JM
004330 =====", : JM
004340 AT (04,10), : JM
004350 "Default options are listed below. They may be changed as", : JM
004360 AT (05,10), : JM
004370 "appropriate for the current document analysis.", : JM
004380 AT (07,19), : JM
004390 "1.) Report Reading Grade Level....(Y/N)", : JM
004400 AT (07,59), RGL$ , CH(01), : JM
004410 AT (08,19), : JM
004420 "2.) Count Prepositions.....(Y/N)", : JM
004430 AT (08,59), PREP_CNT$ , CH(01), : JM
004440 AT (09,19), : JM
004450 "3.) Flag Passive Voice.....(Y/N)", : JM
004460 AT (09,59), PASSV$ , CH(01), : JM
004470 AT (10,19), : JM
004480 "4.) Flag Long Sentences.....(Y/N)", : JM
004490 AT (10,59), LNG_SEN$ , CH(01), : JM
004500 AT (11,19), : JM
004510 "5.) Flag Long Paragraphs.....(Y/N)", : JM
004520 AT (11,59), LNG_PAR$ , CH(01), : JM
004530 AT (12,23), : JM
004540 "If YES, How Long?", : JM
004550 AT (12,42),"Words.....", : JM

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 13:49 12/14/84 Page 9  
Input File is ANALYSIS in Library CRES@SRC on Volume CRES01

```

004560 AT (12,59), FAC(HEX(82)), LNGWDS$, CH(03), RANGE("001","999"), ! JM
004570 AT (13,42), "Sentences.....", ! JM
004580 AT (13,59), FAC(HEX(82)), LNGSNT$, CH(02), RANGE("01","99"), ! JM
004590 AT (14,19), ! JM
004600 "6.) Flag Double Negatives.....(Y/N)", ! JM
004610 AT (14,59), NEG$, CH(01), ! JM
004620 AT (15,19), ! JM
004630 "7.) Flag & List Uncommon Words....(Y/N)", ! JM
004640 AT (15,59), UNCOM$, CH(01), ! JM
004650 AT (16,19), ! JM
004660 "8.) Include Substitutes.....(Y/N)", ! JM
004670 AT (16,59), NUM_SUB$, CH(01), ! JM
004680 AT (17,19), ! JM
004690 "9.) Interim Analyses.....(Y/N)", ! JM
004700 AT (17,59), INT_ANL$, CH(01), ! JM
004710 AT (18,18), ! JM
004720 "10.) Number of Pages Between Reports....", ! JM
004730 AT (18,59), FAC(HEX(82)), NP$, CH(02), RANGE("01","99"), ! JM
004740 AT (19,18), ! JM
004750 "11.) Check Personal Sentences.....(Y/N)", ! JM
004760 AT (19,59), PERS_SENT$, CH(01), ! JM
004770 AT (20,18), ! JM
004780 "12.) Check Truncated Passive Voice.(Y/N)", ! JM
004790 AT (20,59), TRNK_PV$, CH(01), ! JM
004800 AT (22,27), ! JM
004810 "Press", ! JM
004820 AT (22,33), FAC(HEX(84)), ENTER$, ! JM
004830 AT (22,39), ! JM
004840 "to continue ", ! JM
004850 AT (23,20), ! JM
004860 "Press", ! JM
004870 AT (23,26), FAC(HEX(84)), PF15$, ! JM
004880 AT (23,31), ! JM
004890 "to return to previous menu ", ! JM
004900 AT (24,18), ! JM
004910 "Press", ! JM
004920 AT (24,24), FAC(HEX(84)), PF$, ! JM
004930 AT (24,29), ! JM
004940 "to return to CRES Master Menu", ! JM
004950 KEYS(HEX(0010200F1F)), KEY(REC%) JM
004960 IF REC%=16 OR REC%=32 THEN ENDPRG JM
004970 IF REC%=15 OR REC%=31 THEN SCREEN1 JM
004980 IF LNG_PAR$="Y" THEN CONVERT LNGWDS$ TO LNGWDS% JM
004990 IF LNG_PAR$="Y" THEN CONVERT LNGSNT$ TO LNGSNT% JM
005000 NUM_SUBZ$=NUM_SUB$ JM
005010 IF NUM_SUB$="Y" THEN NUM_SUB$="2" ELSE NUM_SUB$="0" JM
005020 CONVERT NP$ TO NP% /* ANALYSIS USES NUMERIC */ JM
005030 BP%=NP% :GP%=NP% /* SET UP BREAK POINT */ JM
005040 JM
005050 JM
005060 /* FIRST, LET'S STORE DEFAULTS IF NECESSARY */ JM
005070 JM
005080 IF RE$<>"1." THEN OPEN_DOC1 JM
005090 CALL "KIKOFF" ADDR(RGL$,PREP_CNT$,PASSV$,LNG_SEN$,LNG_PAR$, ! JM
005100 LNGWDS$,LNGSNT$,NEG$,UNCOM$,NUM_SUB$,INT_ANL$,NP$,ADVFN$, ! JM
005110 PERS_SENT$,LNG_CLS$,TRNK_PV$,TECFIL$(),TECLIB$(),TECVOL$(),RE$, ! JM
005111 INLIB$,INVOL$) JM

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 13:49 12/14/84 Page 10  
Input File is ANALYSIS in Library CRES@SRC on Volume CRES01

```

005120 J M
005130 /* open the document to be analyzed on #1 for input */ J M
005140 J M
005150 OPEN_DOC1: IF OPEN%(1) = 1 THEN CALL "WPCLOSE" ADDR(RET%,#1) J M
005160 SELECT WS J M
005170 ID$=STR(DOCFIL$,1,4) & LIB$ J M
005180 . CALL "WPOPEN" ADDR (RET%,#1,ID$,"I",0%,DOCFIL$,DOCLIB$,DOCVOL$) J M
005190 J M
005200 IF RET%=0 THEN MORE J M
005210 PRINT AT (22,1);"Error on WPOPEN #1 " J M
005220 GOTO IO_ERR1 J M
005230 J M
005240 MORE: OPEN%(1) = 1 J M
005250 CALL "WPSTART" ADDR (RET%,#1,1%,2%) J M
005260 IF RET% = 0 THEN AOK1 J M
005270 PRINT AT (22,1);"Error on WPSTART - First #1" J M
005280 GOTO IO_ERR1 J M
005290 J M
005300 /* TEMPORARY ASSIGNMENT FOR INLIB$ */ J M
005310 J M
005320 AOK1: J M
005330 IF OPEN%(2) = 1 THEN CLOSE #2 J M
005340 OPEN NOGETPARM #2, SHARED, FILE=COMFIL$, : J M
005350 LIBRARY=COMLIB$, VOLUME=COMVOL$ J M
005360 OPEN%(2) = 1 J M
005370 J M
005380 IF OPEN%(3) = 1 THEN CLOSE #3 J M
005390 OPEN NOGETPARM #3, SHARED, FILE=WRDFIL$, : J M
005400 LIBRARY=WRDLIB$, VOLUME=WRDVOL$ J M
005410 OPEN%(3) = 1 J M
005420 J M
005430 IF OPEN%(4) = 1 THEN CLOSE #4 J M
005440 OPEN NOGETPARM #4, SHARED, FILE=PHRFIL$, : J M
005450 LIBRARY=PHRLIB$, VOLUME=PHRVOL$ J M
005460 OPEN%(4) = 1 J M
005470 J M
005480 FOR I% = 1 TO 7 J M
005490 IF TECFIL$(I%)="" " THEN NTI J M
005500 J% = I% + 4 J M
005510 IF OPEN%(J%) = 1 THEN CLOSE #J% J M
005520 OPEN NOGETPARM #J%, SHARED, FILE=TECFIL$(I%), : J M
005530 LIBRARY=TECLIB$(I%), VOLUME=TECVOL$(I%) J M
005540 OPEN%(J%) = 1 J M
005550 NTI: NEXT I% J M
005560 J M
005570 PRINTFL$="ANALYS##" : I%=0 J M
005580 C: CONVERT I% TO STR(PRINTFL$,7,2), PIC(00) J M
005590 CALL "LIMITS" (PRINTFL$,PLIB$,PVOL$,RC$,MSG$) J M
005600 I%=I%+1 J M
005610 IF RC$ = "Y" THEN C J M
005620 J M
005630 IF OPEN%(12) = 1 THEN CLOSE #12 J M
005640 OPEN NOGETPARM #12, OUTPUT, FILE=PRINTFL$, : J M
005650 LIBRARY=PLIB$, VOLUME=PVOL$ J M
005660 OPEN%(12) = 1 J M
005670 J M
005680 ABBRE%=0 J M

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 13:49 12/14/84 Page 11  
Input File is ANALYSIS in Library CRES@SRC on Volume CRES01

```

005690 CALL "LIMITS"("ABBRELST",INLIB$,INVOL$,RC$,MSG$) JM
005700 IF RC$ <> "Y" THEN SK JM
005730 JM
005740 OPEN NOGETPARM #13, SHARED, FILE="ABBRELST", : JM
005750 LIBRARY= INLIB$ , VOLUME = INVOL$ JM
005770 READ #13, IOBUF$ /* GETS RID OF ABBREVIATION HEADER */ JM
005780 I%=1 JM
005790 REED: READ #13, IOBUF$, EOD GOTO CONT JM
005800 $UNPACK (F=FI$) IOBUF$ TO ABBRE$(I%) JM
005810 I%=I%+1 JM
005820 GOTO REED JM
005830 JM
005840 CONT: CLOSE #13 JM
005850 ABBRE%=I%-1 /* SET ABBREVIATION ARRAY LENGTH */ JM
005860 SK:OPEN NOGETPARM #14, INPUT, FILE="ELIMSPCH", LIBRARY=INLIB$, : JM
005870 VOLUME=INVOL$ JM
005880 JM
005890 READ #14, STR(TABLE$( ),,256), EOD GOTO IO_ERR9 JM
005900 JM
005910 CLOSE #14 JM
005920 JM
005930 GOSUB ANALYZE /* ANALYZE TEXT FILES */ JM
005940 JM
005950 GOSUB CLSFL /* CLOSE ALL FILES */ JM
005960 JM
005970 GOSUB ANLDISP /* DISPLAY OR PRINT OUTPUT SCREEN */ JM
005980 IF HOME%=1 THEN ENDPRG /* END THE PROGRAM NOW */ JM
005990 GOTO SCRIN JM
006000 JM
006010 /***** JM
006020 /* JM
006030 /* THIS SUBROUTINE GENERATES A SCREEN FOR THE USER TO DECIDE */ JM
006040 /* HOW HE/SHE WOULD LIKE TO SEE THE GENERATED OUTPUT. */ JM
006050 /* JM
006060 /***** JM
006070 ANLDISP: JM
006080 ER$=" " JM
006090 PF1$="PF1" JM
006100 CWMENU: JM
006110 ACCEPT : JM
006120 AT (01,03), : JM
006130 "CRES: Document Analysis Subsystem (Display of Ana: JM
006140 lysis Screen)", : JM
006150 AT (02,03), : JM
006160 "*****: JM
006170 *****", : JM
006180 AT (05,14), : JM
006190 "CRES Analysis as specified has been completed. Your", : JM
006200 AT (06,14), : JM
006210 "output file is in the Print Queue and named as follows:", : JM
006220 AT (10,12), : JM
006230 "FILE:", : JM
006240 AT (10,18), FAC(HEX(84)), PRINTFL$, CH(08), : JM
006250 AT (10,33), : JM
006260 "LIBRARY:", : JM
006270 AT (10,42), FAC(HEX(84)), PLIB$, CH(08), : JM
006280 AT (10,57), : JM

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 13:49 12/14/84 Page 12  
Input File is ANALYSIS in Library CRES@SRC on Volume CRES01

```

006290 "VOLUME:", : JM
006300 AT (10,65), FAC(HEX(84)), PVOL$, CH(06), : JM
006310 AT (13,02), FAC(HEX(94)), ER$, : JM
006320 AT (16,21), : JM
006330 "Press", : JM
006340 AT (16,27), FAC(HEX(84)), PF1$, CH(03), : JM
006350 AT (16,31), : JM
006360 "to display file at terminal", : JM
006370 AT (17,21), : JM
006380 "Press", : JM
006390 AT (17,27), FAC(HEX(84)), PF15$, CH(04), : JM
006400 AT (17,32), : JM
006410 "to analyze another text", : JM
006420 AT (18,21), : JM
006430 "Press", : JM
006440 AT (18,27), FAC(HEX(84)), PF$, CH(04), : JM
006450 AT (18,32), : JM
006460 "to return to CRES Master Menu", : JM
006470 KEYS(HEX(0110F1F1020)),KEY(REC%) JM
006480 IF REC%=15 OR REC%=31 THEN RETURN JM
006490 IF REC%=16,OR REC%=32 THEN HOME%=1 JM
006500 IF HOME%=1 THEN RETURN JM
006510 JM
006520 CLOSE WS JM
006530 MSG2$ = "CANCEL Processing" JM
006540 MSK$=HEX(00) JM
006550 CMSG$ = " " JM
006560 CALL "LINK" ADDR("ANALDISP","P",RUNLIB$,RUNVOL$,3%, : JM
006570 PRINTFL$,PLIB$,PVOL$,"C",MSG2$,27%,"H",MSK$,CMSG$, : JM
006580 255%,C%,R%) JM
006590 ER$ = " " JM
006600 IF C%=0 AND R%=0 THEN RETURN JM
006610 IF C%<>0 OR R%<>0 : JM
006620 THEN ER$ = STR(CMSG$,141) JM
006630 GOTO CWMENU JM
006640 /***** JM
006650 /* JM
006660 /* THIS SUBROUTINE CLOSES ANY FILES THAT ARE STILL OPEN. JM
006670 /* JM
006680 /***** JM
006690 CLSFL: JM
006700 FOR I%=2 TO 13 JM
006710 IF OPEN%(I%) = 1 THEN CLOSE #I% JM
006720 NEXT I% JM
006730 IF OPEN%(1) = 1 THEN CALL "WPCLOSE" ADDR(RET%,#1) JM
006740 MAT OPEN%=ZER JM
006750 RETURN JM
006760 JM
006770 ENDPRG: GOSUB CLSFL : END /* END THE PROGRAM NOW */ JM
006780 JM
006790 IO_ERR1: PRINT AT(23,1), "I/O ERROR ON DOCUMENT FILE, #1. FILE";! JM
006800 " STATUS IS : ";RET%; "AND "; FS(#1) JM
006810 STOP "PRESS ENTER TO CONTINUE" JM
006820 GOSUB CLSPL : GOTO SCRIN JM
006830 JM
006840 IO_ERR2: PRINT AT(23,1), "I/O ERROR ON COMMON WORD LIST FILE, #2"! JM
006850 ;". FILE STATUS IS : ";FS(#2) JM

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 13:49 12/14/84 Page 13  
Input File is ANALYSIS in Library CRES@SRC on Volume CRES01

```

006860          STOP "PRESS ENTER TO CONTINUE" JM
006870          GOSUB CLSFL : GOTO SCRN JM
006880 JM
006890 IO_ERR3: PRINT AT(23,1), "I/O ERROR ON WORD SUBST. LIST FILE, #3": JM
006900          ;". FILE STATUS IS : "; FS(#3) JM
006910          STOP "PRESS ENTER TO CONTINUE" JM
006920          GOSUB CLSFL : GOTO SCRN JM
006930 JM
006940 IO_ERR4: PRINT AT(23,1), "I/O ERROR ON PHRASE SUBST. LIST FILE"; ! JM
006950          " #4. FILE STATUS IS : ";FS(#4) JM
006960          STOP "PRESS ENTER TO CONTINUE" JM
006970          GOSUB CLSFL : GOTO SCRN JM
006980 JM
006990 IO_ERR5: PRINT AT(23,1), "I/O ERROR ON TECH. WORD LIST FILE, #5";! JM
007000          ". FILE STATUS IS : ";FS(#5) JM
007010          STOP "PRESS ENTER TO CONTINUE" JM
007020          GOSUB CLSFL : GOTO SCRN JM
007030 JM
007040 IO_ERR6: PRINT AT(23,1), "I/O ERROR ON TECH. WORD LIST FILE, #6";! JM
007050          ". FILE STATUS IS : ";FS(#6) JM
007060          STOP "PRESS ENTER TO CONTINUE" JM
007070          GOSUB CLSFL : GOTO SCRN JM
007080 JM
007090 IO_ERR7: PRINT AT(23,1), "I/O ERROR ON TECH. WORD LIST FILE, #7";! JM
007100          ". FILE STATUS IS : ";FS(#7) JM
007110          STOP "PRESS ENTER TO CONTINUE" JM
007120          GOSUB CLSFL : GOTO SCRN JM
007130 JM
007140 IO_ERR8: PRINT AT(23,1), "I/O ERROR ON PRINT FILE, #12. FILE ST";! JM
007150          "ATUS IS : ";FS(#12) JM
007160          STOP "PRESS ENTER TO CONTINUE" JM
007170          GOSUB CLSFL : GOTO SCRN JM
007180 JM
007190 IO_ERR9: PRINT AT(23,1), "I/O ERROR ON TABLE FILE, #14. FILE ST";! JM
007200          "ATUS IS : ";FS(#14) JM
007210          STOP "PRESS ENTER TO CONTINUE" JM
007220          GOSUB CLSFL : GOTO SCRN JM
007230 JM
007240 IO_ERR10:PRINT AT(23,1), "I/O ERROR ON TECH. WORD LIST FILE, #8";! JM
007250          ". FILE STATUS IS : ";FS(#8) JM
007260          STOP "PRESS ENTER TO CONTINUE" JM
007270          GOSUB CLSFL : GOTO SCRN JM
007280 JM
007290 IO_ERR11:PRINT AT(23,1), "I/O ERROR ON TECH. WORD LIST FILE, #9";! JM
007300          ". FILE STATUS IS : ";FS(#9) JM
007310          STOP "PRESS ENTER TO CONTINUE" JM
007320          GOSUB CLSFL : GOTO SCRN JM
007330 JM
007340 IO_ERR12:PRINT AT(23,1),"I/O ERROR ON TECH. WORD LIST FILE, #10";! JM
007350          ". FILE STATUS IS : ";FS(#10) JM
007360          STOP "PRESS ENTER TO CONTINUE" JM
007370          GOSUB CLSFL : GOTO SCRN JM
007380 JM
007390 IO_ERR13:PRINT AT(23,1),"I/O ERROR ON TECH. WORD LIST FILE, #11";! JM
007400          ". FILE STATUS IS : ";FS(#11) JM
007410          STOP "PRESS ENTER TO CONTINUE" JM
007420          GOSUB CLSFL : GOTO SCRN JM

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 13:49 12/14/84 Page 14  
Input File is ANALYSIS in Library CRES@SRC on Volume CRES01

```

007430 JM
007440 IO_ERR14:PRINT AT(23,1), "I/O ERROR ON ABBREVIATION FILE, #13": JM
007450 " . FILE STATUS IS : ";PS(#13) JM
007460 STOP "PRESS ENTER TO CONTINUE" JM
007470 GOSUB CLSFL : GOTO SCRIN JM
007480 JM
007490 /***** JM
007500 /* JM
007510 /* THIS SUBROUTINE ANALYZES A DOCUMENT FILE AND OUTPUTS */ JM
007520 /* THE RESULT IN A PRINT FILE. SEVERAL OPTIONAL WORD LISTS MAY */ JM
007530 /* BE USED IN THE ANALYSIS. */ JM
007540 /* JM
007550 /***** JM
007560 JM
007570 ANALYZE: JM
007580 JM
007590 /* DECLARE MORE VARIABLES */ JM
007600 JM
007610 DIM ILINE$(1)80, WORDS$20, RECORD$40,PREC$70,REC1$80,BLANK$1 JM
007620 DIM OLINE$77, WORDS$80, WORD1$24, BB$1, BC$1, BS$1, AS1 JM
007630 DIM INVLD$1, CW$1, FOUND$1, CLSPOS$1, T1$1, T2$1,T3$1,E_O F$1 JM
007640 DIM SENTPOS$1, NEG$(3), PERSENT$1, END_O_PAR$1, WORDS$(2)160 JM
007650 DIM Z1$1,Z2$2,Z3$3,UPDMSG$51,WORD2$24,BEGINY$3,ENDY$3,BUFFER$33 JM
007660 DIM SAVSUB$(5)40,SAVSUBL$(5),SAVREC1$(5)80,WORDZ$80 JM
007670 BLANK$="N" JM
007680 MAT SAVSUBL$ = ZER JM
007690 WW1$=0 :WW$=0 /* INITIALIZE UPDATE VARIABLES */ JM
007700 GOSUB UPDATE JM
007710 CALL "TTLPG$" ADDR(BUFFA$(),PAG_TITL$,DOCFIL$,LIB$,DOCVOL$, ! JM
007720 RGL$,PREP_CNT$,PASSV$,LNG_SEN$,LNG_PAR$,LNGWDS$,LNGSNT$, ! JM
007730 NEG$,UNCOM$,NUM_SUBZ$,INT_ANL$,NP$,ADVFN$,PERS_SENT$, ! JM
007740 LNG_CLS$,TRNK_PV$,TECFIL$(),TECLIB$(),TECVOL$(),IE$) JM
007750 JM
007760 GOSUB TTL_PGE JM
007770 JM
007780 GOSUB SETUP /* INITIALIZE PROG. */ JM
007790 JM
007800 CLGW: JM
007810 JM
007820 /* GET NEXT WORD */ JM
007830 GOSUB GETWORD JM
007840 ON (I$+1) GOTO IO_ERR1, WDFND, BLDOP1, BLDOP1, WFO JM
007870 JM
007880 WDFND: /* ANALYZE WORD */ JM
007890 GOSUB CHKWORD JM
007900 IF I$=0 THEN WFO JM
007910 JM
007920 /* CASES OF NO ANALYSIS */ JM
007930 JM
007940 IF NUM_SUB$="2" AND CW$<>"N" THEN CHKPHR JM
007950 IF CW$ = "N" OR UNCOM$ <> "Y" THEN BLDOP1 JM
007960 IF NUM_SUB$<>"2" THEN BLDOP1 JM
007970 JM
007980 /* CHECK PHRASE FILE */ JM
007990 CHKPHR: GOSUB CHKPHSFL JM
008000 IF I$=0 THEN WFO JM
008010 IF CW$<>"P" THEN BLDOP1 /* NO PARTIAL PHRASE FOUND */ JM

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 13:49 12/14/84 Page 15  
Input File is ANALYSIS in Library CRES@SRC on Volume CRES01

```

008020                                     JM
008030      /* MUST HAVE FOUND PARTIAL PHRASE - SAVE IT AND TRY AGAIN */      JM
008040                                     JM
008050      SAVPH$(PHCT%)=STR(WORDS$,1,WLEN%)      /* SAVE THE WORD */ JM
008060      SAVPHL$(PHCT%)=WLEN%      /* SAVE THE WORD'S LENGTH */ JM
008070      SAVSYL$(PHCT%)=TLC%      /* SAVE THE SYLLABLE COUNT */ JM
008080      IF WSL$<>"Y" THEN PHCT%=PHCT%+1      /* INCREMENT PHCT% */ JM
008090      IF WSL$<>"Y" THEN CLGW      /* NO NEED TO SAVE SUBS. */ JM
008100      SAVRECL$(PHCT%)=RECL%      /* SAVE THE RECORD */ JM
008110                                     JM
008120      /* SINCE THIS WORD SUBSTITUTE COULD BE TAKEN CARE OF IN SEVERAL */ JM
008130      /* PLACES IN THIS PROGRAM, WE NEEDED TO SAVE IT TWO WAYS - AS A */ JM
008140      /* RECORD (SAVRECL%) AND AS COMPONENTS (SAVSUB$ & SAVSUBL%) */ JM
008150                                     JM
008160      K1%=-1      JM
008170      K1%=POS(STR(RECL$,21,20)=" ")      JM
008180      SUBZ: IF STR(RECL$,K1%+21,1)=" " OR K1%=20 THEN CNT1      JM
008190      K4%=POS(STR(RECL$,K1%+21,20-K1%)=" ")      /* GET THE REST */ JM
008200      IF K4%=0 THEN K1%=20      /*THE SUBSTITUTE TOOK 20 SPACES*/ JM
008210      K1%=K1%+K4%      JM
008220      GOTO SUBZ      JM
008230      CNT1: K1%=MIN(20,K1%)      JM
008240      J%=K1%+5      /* INCLUDES "<" "....">" */ JM
008250      JM
008260      /* ANOTHER SUBSTITUTE ? */      JM
008270      JM
008280      IF STR(RECL$,41,1)=" " THEN J%=J%+2      JM
008290      IF STR(RECL$,41,1)=" " THEN CNT2      JM
008300      K2%=0      JM
008310      K2%=POS(STR(RECL$,41,15)=" ")      JM
008320      SUB1: IF STR(RECL$,K2%+41,1)=" " OR K2%=15 THEN CNT2      JM
008330      K4%=POS(STR(RECL$,K2%+41,15-K2%)=" ")      /* GET THE REST */ JM
008340      IF K4%=0 THEN K2%=15      /* THE SUBSTITUTE TOOK 15 SPACES */ JM
008350      K2%=K2%+K4%      JM
008360      GOTO SUB1      JM
008370      CNT2: K2%=MIN(15,K2%)      JM
008380      IF K2%>0 THEN K2% = K2% + 3      /* INCLUDES "/" & TWO SPACES */ JM
008390      SAVSUBL$(PHCT%)=J%+K2%      JM
008400      SAVSUB$(PHCT%)="<" & STR(RECL$,21,K1%)      JM
008410      IF K2%<>0 THEN SAVSUB$(PHCT%)=SAVSUB$(PHCT%) & "/" &      ! JM
008420      STR(RECL$,41,K2%) & ">"      ! JM
008430      ELSE SAVSUB$(PHCT%)=SAVSUB$(PHCT%) & ">"      JM
008440      WSL$="N"      /* RESET THE FLAG */ JM
008450      PHCT% = PHCT% + 1      /* INCREMENT THE COUNT OF WORDS HELD */ JM
008460      GOTO CLGW      /* GET THE NEXT WORD */ JM
008470      JM
008480      /* BUILD OUTPUT LINE */      JM
008481      JM
008490      BLDOP1:      JM
008500      I%=1      JM
008510      GOSUB SELOPT      JM
008520      IF I% <> 0 THEN CLGW      JM
008530      JM
008540      /* WRITE FINAL OUTPUT */      JM
008541      JM
008550      WFO:      JM
008560      E_O_F$="Y"      JM

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 13:49 12/14/84 Page 16  
Input File is ANALYSIS in Library CRES@SRC on Volume CRES01

```

008561 IF UNCOM$="Y" AND UNCPT% 1 THEN GOSUB ADD_UNCWL JM
008570 CW$="N" /* NO PHRASE */ JM
008580 END_O_PAR$="Y" /* END OF PARAGRAPH */ JM
008590 I% = 1 JM
008600 GOSUB SELOPT JM
008610 IF I% = 0 THEN ENDSUB JM
008620 JM
008630 /* COMPLETE PRINTING THE LAST LINE IF NECESSARY */ JM
008640 JM
008650 IF IO% <= 1 THEN INTY /* PRINT THE REMAINDER OF LINE */ JM
008660 JM
008670 GOSUB PRTLNE JM
008680 IF I% = 0 THEN ENDSUB JM
008690 JM
008700 /* CHECK ON NEED FOR RGL */ JM
008710 /* NO WORDS READ OR DON'T WANT INTERIM RGL */ JM
008720 JM
008730 INTY: IF WC% = 0 OR INT_ANL$ <> "Y" THEN CHKRGL JM
008740 IF PC% - 1 <= BP% THEN CIRCLE /* NO INTERIM ANALYSIS NEEDED */ JM
008750 GOSUB PRTRGL JM
008760 IF IQ% = 0 THEN ENDSUB JM
008770 JM
008780 IF IDX% = 0 OR UNCOM$ <> "Y" THEN CHKRGL /* NO UNCOMMON WORDS */ JM
008790 JM
008800 GOSUB UWL JM
008810 IF IU% = 0 THEN ENDSUB JM
008820 JM
008830 /* DO WE PRINT SUMMARY RGL */ JM
008840 JM
008850 CHKRGL: JM
008860 IF E_O_F$ = "Y" THEN E_O_F$ = "E" JM
008870 IF E_O_F$ <> "E" THEN RETURN JM
008880 IF UNCOM$ = "Y" AND IDX% <> 0 THEN GOSUB UWL JM
008890 IF RGL$ <> "Y" THEN RETURN JM
008900 LAST$ = "Y" /* USED TO SET PROPER HEADING ON PRINTOUT */ JM
008910 IF INT_ANL$ <> "Y" THEN CIRCLE /* NOT BEEN KEEPING SUBTOTALS */ JM
008920 LS% = XLS% JM
008930 SC% = XSC% JM
008940 SLC% = XSLC% JM
008950 UW% = XUC% JM
008960 WC% = XWC% JM
008970 CIRCLE: IF RGL$ <> "Y" THEN CI JM
008980 LAST$ = "Y" JM
008990 GOSUB PRTRGL JM
009000 CI: IF PC% - 1 <= BP% AND UNCOM$ = "Y" AND IDX% <> 0 THEN GOSUB UWL JM
009010 GOTO ENDSUB JM
009020 JM
009030 /******/ JM
009040 /* */ JM
009050 /* THIS SUBROUTINE INITIALIZES VARIABLES FOR THE ANALYSIS. */ JM
009060 /* */ JM
009070 /******/ JM
009080 JM
009090 SETUP: JM
009100 UNC$(), WORDS$(), ILINE$(), OLINE$, SAVPH$() = ALL(HEX(20)) JM
009110 PERSENT$, END_O_PAR$, INVLD$, CW$, T1$, T2$, FOUND$, CLSPOS$, T3$ = " JM
009120 SENTPOS$ = "B" : E_O_F$ = "N" JM

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 13:49 12/14/84 Page 17  
Input File is ANALYSIS in Library CRES@SRC on Volume CRES01

```

009130 MAT NEG%=ZER :MAT SAVPHL%=ZER :MAT UCWCT%=ZER JM
009140 MAT LCLSFLL%=ZER /* SET LONG CLAUSE FLAG TO BLANK */ JM
009150 UCWFLG$()=" " /* SET ALL UCW FLAGS TO BLANK */ JM
009160 UCW$()=" " /* SET ALL UCW ARRAY TO BLANK */ JM
009170 IDX%=0 /* SET THE INDEX TO THE UCW ARRAY TO 0 */ JM
009180 CONSIS%, LIST%, EOS%, PASSV%, TPASSV%=0 JM
009190 WC%, SC%, SLCL%, UW%, LS%, WS%, PS%, TLC%, NC%, PE%, WP%, SP% =0 JM
009200 WCL%, WPL% =0 JM
009210 SLCL% =0 JM
009220 UNCPT%, PHCT%, IO%, PC%=1 /* PC IS PAGE COUNT */ JM
009230 BEGINY$=" 1" /* USED FOR PRINTING RGL */ JM
009240 LC%=0 /* LINE COUNT */ JM
009250 LP%=50 /* LINES / PAGE */ JM
009260 LL%=77 /* CHARS. / LINE */ JM
009270 IL%=257 /* SET SO THAT FIRST LINE WILL BE READ */ JM
009280 WW%, I%=1 JM
009310 JM
009320 RETURN /* SETUP SUBROUTINE RETURN */ JM
009330 JM
009340 /***** JM
009350 /* JM
009360 /* THIS SUBROUTINE GETS THE NEXT WORD TO BE ANALYZED. JM
009370 /* JM
009380 /***** JM
009390 JM
009400 GETWORD: JM
009410 I%=0 JM
009420 IL%=MAX(1, IL%) JM
009430 IF IL% > 80 THEN GETLNE JM
009440 JM
009450 GETSTPOS: /* GET THE START POSITION */ JM
009460 J% = POS(STR(ILINE$()), IL%, 81-IL%) > " " JM
009470 IF J%=0 THEN GETLNE JM
009480 JM
009490 IL% = IL% + J% - 1 /* RESET LINE POINTER */ JM
009500 JM
009510 /* GET END POSITION */ JM
009520 J% = POS(STR(ILINE$()), IL%, 81-IL%) = " " JM
009530 IF J% <> 0 THEN FIND JM
009540 W$=STR(ILINE$()), IL%, 81-IL%) JM
009550 CR:GOSUB GETLINE JM
009560 J%=POS(STR(ILINE$()), IL%, 81-IL%) = " " JM
009570 IF J%=1 THEN DONE1 JM
009572 IF J%=0 THEN CR JM
009580 WLEN%=MIN(30, J%-1) /* HAD TO MAKE THIS LARGER THAN 25! */ JM
009590 WORDS$=STR(ILINE$()), IL%, WLEN%) JM
009600 WORDS$=W$ & WORDS$ JM
009610 WLEN%=LEN(WORDS$) JM
009620 GOTO INCR JM
009630 JM
009640 DONE1:WORDS$=W$ JM
009650 WLEN%=LEN(WORDS$) JM
009660 GOTO INCR JM
009670 FIND: JM
009680 WLEN%= MIN(30, J% - 1) /* HAD TO MAKE THIS LARGER THAN 25 */ JM
009690 WORDS$=STR(ILINE$()), IL%, WLEN%) JM
009700 JM

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 13:49 12/14/84 Page 18  
Input File is ANALYSIS in Library CRES@SRC on Volume CRES01

```

009710 INCR: IF WLEN% < 2 THEN ARON /* CAN'T BE @@ */ JM
009720 IF WORDS$ = "==" OR STR(WORDS$,1,2) = "==" THEN GOSUB ANALSKIP JM
009730 IF I% = 4 THEN RET /* EOF FOUND */ JM
009740 ARON: IW% = MIN(5, IW%+1) /* RESET POINTER FOR ARRAY OF WORDS */ JM
009750 IL% = IL% + J% : LW%(IW%) = WLEN% JM
009760 JM
009770 /* CHECK FOR CARRIAGE RETURN ( FLAGS END OF PARAGRAPH ) */ JM
009780 IF STR(WORDS$, WLEN%, 1) = HEX(03) THEN END_O_PAR$ = "Y" JM
009790 I% = 1 /* FLAG A SUCCESS */ JM
009800 GOTO RET JM
009810 JM
009820 /* CALL THE ROUTINE TO GET THE NEXT LINE AND RESET POINTERS */ JM
009830 JM
009840 GETLINE: JM
009850 GOSUB GETLINE JM
009860 IF I% = 1 THEN GETSTPOS JM
009870 RET: JM
009880 RETURN /* GETWORD SUBROUTINE RETURN */ JM
009890 JM
009900 /***** JM
009910 /* JM
009920 /* THIS SUBROUTINE SKIPS ANY TEXT THAT USER SPECIFIES AND DOES */ JM
009930 /* NOT ANALYZE THAT TEXT. */ JM
009940 /* JM
009950 /***** JM
009960 JM
009970 ANALSKIP: /* ANALSKIP START */ JM
009980 IF IO% + WLEN% > LL% THEN GOSUB PRTLINE JM
009990 STR(OLINE$, IO%, WLEN%) = WORDS$ JM
010000 IO% = IO% + WLEN% + 1 JM
010010 I% = 0 JM
010020 QUIT% = 0 /* FLAG TO TELL US WHEN ANOTHER @@ HAS BEEN FOUND */ JM
010030 IL% = MAX(1, IL% + WLEN% + 1) JM
010040 IF IL% > 80 THEN GETLINE1 JM
010050 JM
010060 GETSTPOS1: /* GET THE START POSITION */ JM
010061 J% = 0 JM
010062 IF IL% > 80 THEN 10080 JM
010070 J% = POS(STR(ILINE$(), IL%, 81-IL%)) > " " JM
010080 IF J% = 0 THEN GETLINE1 JM
010090 IL% = IL% + J% - 1 /* RESET LINE POINTER */ JM
010100 JM
010110 /* GET END POSITION */ JM
010120 J% = POS(STR(ILINE$(), IL%, 81-IL%)) = " " JM
010130 IF J% < 0 THEN FIND1 JM
010140 W$ = STR(ILINE$(), IL%, 81-IL%) JM
010150 GOSUB GETLINE JM
010160 J% = POS(STR(ILINE$(), IL%, 81-IL%)) = " " JM
010170 IF J% = 1 THEN DONE2 JM
010180 WLEN% = MIN(30, J% - 1) /* HAD TO MAKE THIS LARGER THAN 25! */ JM
010190 WORDS$ = STR(ILINE$(), IL%, WLEN%) JM
010200 WORDS$ = W$ & WORDS$ JM
010210 WLEN% = LEN(WORDS$) JM
010220 GOTO INCR1 JM
010230 JM
010240 DONE2: WORDS$ = W$ JM
010250 WLEN% = LEN(WORDS$) JM

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 13:49 12/14/84 Page 19  
Input File is ANALYSIS in Library CRES0SRC on Volume CRES01

```

010260 GOTO INCR1 JM
010270 FIND1: JM
010280 WLEN% = MIN(30,J% - 1) /* HAD TO MAKE THIS LARGER THAN 25 */ JM
010290 WORDS$=STR(ILINE$(),IL%,WLEN%) JM
010300 JM
010310 INCR1: IF QUIT%=1 THEN RET1 /* WE'RE READY TO LEAVE */ JM
010320 IF IO%+WLEN%>LL% THEN GOSUB PRTLNE JM
010330 STR(OLINE$,IO%,WLEN%)=WORDS$ JM
010340 IO%=IO%+WLEN%+1 JM
010350 IL%=IL%+J% /* INCREMENT LINE POINTER */ JM
010360 IF STR(WORDS$,WLEN%,1)=HEX(03) THEN GOSUB PRTLNE/* END/PAR*/ JM
010370 IF WLEN%<2 THEN GETSTPOS1 JM
010380 IF (WORDS$<>"@@" AND QUIT%=0) OR : JM
010390 (STR(WORDS$,WLEN%-1,2)<>"@@" AND QUIT%=0) THEN GETSTPOS1 JM
010400 QUIT%=1 /* IF HERE, FOUND @@ */ JM
010410 GOTO GETSTPOS1 JM
010420 JM
010430 GETLINE1: JM
010440 GOSUB GETLINE JM
010450 IF I% = 1 THEN GETSTPOS1 JM
010460 JM
010470 RET1: RETURN /* ANALSKIP END */ JM
010480 JM
010490 /***** JM
010500 */ JM
010510 /* THIS SUBROUTINE FILLS THE BUFFER WITH A LINE OF TEXT. */ JM
010520 */ JM
010530 /***** JM
010540 JM
010550 GETLINE: JM
010560 I%, IL% = 1 /* Assume Success */ JM
010570 ILINE$() = ALL(HEX(20)) JM
010580 CALL "WPREAD" ADDR(RET%,#1,"N",ILINE$(),80%,"L") JM
010582 TRAN( ILINE$(),TABLE$()) JM
010590 IF RET% = 0 OR RET% = 24 THEN GET_RET /* Success */ JM
010600 IF RET% = 28 THEN EOF /* EOF */ JM
010610 PRINT AT (22,1); "Error on WPREAD #1" /* Error */ JM
010620 I% = 0 JM
010630 GOTO GET_RET JM
010640 EOF: I% = 4 /*:STOP "FINISHED" */ JM
010650 GET_RET: JM
010660 RETURN /* GETLINE SUBROUTINE RETURN */ JM
010670 JM
010680 /***** JM
010690 */ JM
010700 /* THIS SUBROUTINE ANALYSES THE WORD. */ JM
010710 */ JM
010720 /***** JM
010730 JM
010740 CHKWORD: JM
010750 WS% = WS% + 1 /* INCREMENT WORDS PER SENTENCE */ JM
010760 IF CW$ <> "P" THEN CW$ = " " JM
010770 LEN% = WLEN% :LW%(1)=LEN% JM
010780 WORD1$ = WORDS$ JM
010790 GOSUB SPUNK /* STRIP PUNC. FROM THE WORD */ JM
010800 IF CLSPOS$ <> "B" THEN SAV_ORGWd JM
010810 TPASSV%, PASSV% = 0 JM

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 13:49 12/14/84 Page 20  
Input File is ANALYSIS in Library CRES@SRC on Volume CRES01

```

010820 JM
010830 SAV_ORGWD: /* SAVE THE ORIG. WORD FOR UNCOM. WORD LIST */ JM
010840 WORD0$ = WORD1$ JM
010850 TRAN( WORD1$, : JM
010860 "AaBbCcDdEeFfGgHhIiJjKkLlMmNnOoPpQqRrSsTtUuVvWwXxYyZz" ) : JM
010870 REPLACING JM
010880 JM
010890 /* P. KINCAID ASKED FOR THIS PROGRAM TO FIND THE PHRASE */ JM
010900 /* 'CONSIST(S) OF' FOLLOWED BY 3 COMMAS IN A LONG SENTENCE */ JM
010910 /* AND FLAG IT BY THE TERM 'LIST?' AFTER THE LONG SENTENCE */ JM
010920 /* FLAG. SINCE 'CONSIST(S) OF' IS NOT IN THE PHRASE SUB. */ JM
010930 /* LIST, I USE THE FLAG CONSI% TO HELP ME FIND IT. */ JM
010940 /* CONSI%=0 IMPLIES PHRASE NOT FOUND; CONSI%=1 IMPLIES */ JM
010950 /* PHRASE 'CONSIST(S) OF' FOUND; CONSI%=10 IMPLIES */ JM
010960 /* 'CONSIST(S)' FOUND (NOW LOOK FOR 'OF'); */ JM
010970 JM
010980 IF WORD1$ = "OF" AND CONSI% = 10 THEN CONSI%=1 /* 'LIST?' */ JM
010990 IF CONSI%=10 THEN CONSI%=0 JM
011000 IF STR(WORD1$,1,7)="CONSIST" THEN CONSI%=10 JM
011010 WORD2$=WORD1$ JM
011020 SLASH%=POS(STR(WORD1$,1,LEN%)="/") /* 2 WORDS AND A "/" */ JM
011030 IF SLASH%=0 THEN DOIT /* ONLY ONE WORD */ JM
011040 IF SLASH%=1 AND LEN%=1 THEN DOIT /* ONLY ONE SLASH */ JM
011050 WRDLEN%=LEN% /* SAVE THE ORIGINAL LEN% */ JM
011060 HOLD2$=STR(WORD1$,SLASH%+1,LEN%-SLASH%) /* SAVE 2ND WORD */ JM
011070 LEN%=LEN%-(LEN%-SLASH%+1) /* COMPUTE THE LEN FOR WORD 1 */ JM
011080 STR(WORD1$,SLASH%)=ALL(HEX(20)) /* BLANK OUT 2ND WORD */ JM
011090 GOTO DOIT1 JM
011100 DOIT: SLASH%=POS(STR(WORD1$,1,LEN%)="-") JM
011110 IF SLASH%=0 THEN DOIT1 JM
011120 Z99$=STR(WORD1$,1,LEN%) /* CHECK IF "--" SEPERATES NUMBERS */ JM
011130 A0%=POS(Z99$="0") /* LOOK FOR ZERO */ JM
011140 A1%=POS(Z99$="1") /* LOOK FOR ONE */ JM
011150 A2%=POS(Z99$="2") /* LOOK FOR TWO */ JM
011160 A3%=POS(Z99$="3") /* LOOK FOR THREE */ JM
011170 A4%=POS(Z99$="4") /* LOOK FOR FOUR */ JM
011180 A5%=POS(Z99$="5") /* LOOK FOR FIVE */ JM
011190 A6%=POS(Z99$="6") /* LOOK FOR SIX */ JM
011200 A7%=POS(Z99$="7") /* LOOK FOR SEVEN */ JM
011210 A8%=POS(Z99$="8") /* LOOK FOR EIGHT */ JM
011220 A9%=POS(Z99$="9") /* LOOK FOR NINE */ JM
011230 A10%=A0%+A1%+A2%+A3%+A4%+A5%+A6%+A7%+A8%+A9% JM
011240 IF A10%<>0 THEN SLASH%=0 JM
011250 IF LEN%=1 THEN SLASH%=0 JM
011260 IF SLASH%=0 THEN DOIT1 JM
011270 WRDLEN%=LEN% /* SAVE THE ORIGINAL LEN% */ JM
011280 HOLD2$=STR(WORD1$,SLASH%+1,LEN%-SLASH%) /* SAVE 2ND WORD */ JM
011290 LEN%=LEN%-(LEN%-SLASH%+1) /* COMPUTE THE LEN FOR WORD 1 */ JM
011300 STR(WORD1$,SLASH%)=ALL(HEX(20)) /* BLANK OUT 2ND WORD */ JM
011310 DOIT1: GOSUB SYLLY JM
011320 IF SLASH%=0 THEN ARND /* ONLY 1 WORD */ JM
011330 LEN%=WRDLEN%-SLASH% /* COMPUTE 2ND WORD LEN% */ JM
011340 WORD1$=HOLD2$ /* FILL IN 2ND WORD IN VARIABLE */ JM
011350 SYLCT%=TLC% /* SAVE 1ST WORD SYL. COUNT */ JM
011360 GOSUB SYLLY JM
011370 LEN%=WRDLEN% /* RESTORE WORD LENGTH */ JM
011380 TLC%=TLC%+SYLCT% /* ADD TWO WORDS' SYL COUNT */ JM

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 13:49 12/14/84 Page 21  
Input File is ANALYSIS in Library CRES@SRC on Volume CRES01

```

011390 WORD1$=WORD2$ /* RESTORE THE ORIGINAL WORD */ JM
011400 ARND: JM
011440 IF WORD1$="-" OR WORD1$="/" THEN TLC%=0 /* NO SYLLABLES */ JM
011450 WW%=WW%+1 /* KEEP # OF WORDS PROCESSED FOR PROGRESS SCREEN */ JM
011460 IF WW%=250 THEN GOSUB UPDATE /* REPORT PROGRESS TO USER */ JM
011461 IF STR(WORD0$,1,1)>"A" AND STR(WORD0$,1,1)<="Z" AND : JM
011462 SENTPOS$<>"B" THEN SKPR JM
011470 /* LOOK FOR "N'T" */ JM
011480 IF LEN% < 5 THEN LOOK JM
011490 IF STR(WORD1$,LEN%-2,3) <> "N'T" : JM
011500 THEN LOOK JM
011510 INVLD$ = " " /* WORD OK SO FAR */ JM
011520 NEG%(1+NEG%(3)) = NEG%(1+NEG%(3)) + 1 JM
011530 JM
011540 LOOK: /* LOOK UP WORD IN WORD SUBSTITUTION LIST */ JM
011550 POUND$ = " " JM
011560 IF INVLD$ = "Y" OR (INVLD$ = "P" AND SENTPOS$ <> "E") : JM
011570 THEN SKPR JM
011580 RECORD$ = STR(WORD1$,1,LEN%) JM
011600 TECH% = 0 JM
011610 READ #3, KEY = STR(RECORD$,1,20), REC1$, EOD GOTO CWL JM
011620 Q$ = STR(REC1$,56,1) JM
011630 GOTO SAVSUBS JM
011640 CWL: READ #2, KEY = STR(RECORD$,1,20), RECORD$, EOD GOTO CHEKTECH JM
011650 Q$ = STR(RECORD$,21,1) JM
011660 GOTO GOOD JM
011670 JM
011680 CHEKTECH: /* SEE IF WORD IS IN THE SELECTED TECH. WORD LISTS */ JM
011690 FOR J% = 5 TO 11 JM
011700 IF TECFIL$(J%-4)<> " " THEN : JM
011710 READ #J%, KEY = STR(RECORD$,1,20),RECORD$,EOD GOTO ANOTHA! JM
011720 ELSE ANOTHA JM
011730 Q$ = STR(RECORD$,21,1) JM
011740 TECH% = 1 : J% = 12 JM
011750 ANOTHA: NEXT J% JM
011760 IF TECH% = 0 THEN UNCWD ELSE GOOD JM
011770 JM
011780 SKPR: /* INVALID WORD, SKIP PROCESSING */ JM
011790 CW$ = "N" JM
011800 INVLD$ = "Y" JM
011810 IF SLASH%>=1 THEN UNCWD /* "/" OR "-" IN WORD; PUT IN UNCOM.*/ JM
011820 GOTO CHKLCLS JM
011830 JM
011840 SAVSUBS: /* FOUND WORD IN WORD SUBSTITUTION LIST*/ JM
011850 IF NUM_SUB$="0" THEN CHKLCLS /* NO NEED FOR SUBSTITUTES */ JM
011860 A$=STR(REC1$,56,1) /* FETCH THE PART OF SPEECH */ JM
011870 WSL$="Y" /* SET FLAG TO FETCH SUBSTITUTES IN SELOPT */ JM
011880 RECORD$ = STR(RECORD$,1,20) & STR(REC1$,56,3) JM
011900 GOTO CHK JM
011910 UNCWD: /* SAVE UNCOMMON WORD */ JM
011920 IF INVLD$ = "P" THEN SKPR /* REALLY INVALID */ JM
011930 IF ABBRE%=0 THEN SKP /* CHECK IF IN OUR ABBRE/MNEMONIC LIST */ JM
011940 POUND%=0 JM
011950 FOR QD%=1 TO ABBRE% JM
011960 IF STR(WORD1$,1,LEN%)>STR(ABBRE$(QD%),1,LEN%) THEN QD%=ABBRE% JM
011970 IF STR(WORD1$,1,LEN%)<>STR(ABBRE$(QD%),1,LEN%) THEN DRP JM
011980 IF LEN(ABBRE$(QD%))>LEN%+1 THEN DRP /*TAKE CARE OF '.' LEN */ JM

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 13:49 12/14/84 Page 22  
Input File is ANALYSIS in Library CRES@SRC on Volume CRES01

```

011990 FOUN% = 1 JM
012000 QD% = ABBRE% JM
012010 DRP: NEXT QD% JM
012020 IF FOUN% = 1 THEN SKPR JM
012030 SKP: CW$ = "N" /* PHRASE NOT POSSIBLE */ JM
012040 IF INVLD$ = "Y" THEN JUMP JM
012050 IF UNCOM$ <> "Y" THEN GOOD JM
012060 JUMP: IF UNCT% < 21 THEN UNC$(UNCT%) = WORDO$ /* ROOM FOR 20 */ JM
012070 UNCT% = UNCT% + 1 JM
012080 IW% = MAX(IW%, PHCT%) JM
012090 UCWFLG$(IW%) = "Y" JM
012100 UW% = UW% + 1 /* COUNT UNCOMMON WORDS */ JM
012110 GOTO CHKLCLS JM
012120 JM
012130 GOOD: /* GOOD WORD SO FAR, EVALUATE ATTRIBUTES */ JM
012140 FOUND$ = "Y" JM
012150 AS$ = STR(RECORD$, 21, 1) JM
012160 CHK: IF (AS$ = "P" OR AS$ = "A" OR AS$ = "a" ) AND CLSPOS$ = "B" ! JM
012170 THEN CLSPOS$ = "Y" JM
012180 IF AS$ = "V" AND CLSPOS$ = "B" THEN CLSPOS$ = "Y" JM
012190 IF CLSPOS$ = "Y" THEN NC% = NC% + 1 /* INC. # WORDS/CLAUSE */ JM
012200 IF AS$ <> "C" THEN SYLC JM
012210 CLSPOS$ = "E" /* SET END OF CLAUSE */ JM
012220 NEG$(1), NEG$(2), PASSV%, TPASSV% = 0 JM
012230 JM
012240 SYLC: /* CHECK SYLLABLE COUNT */ JM
012250 JM
012260 /* MANDATORY TECHNICAL WORDS COUNT AS 1 SYLLABLE! */ JM
012270 JM
012280 IF TECH% = 1 AND STR(RECORD$, 27, 1) = "" THEN TLC% = 1 JM
012290 JM
012300 /* WE'RE CHECKING FOR 'BY' IMMEDIATELY FOLLOWING OR IMMEDIATELY */ JM
012310 /* PRECEDING 'TO BE' AND A PAST PARTICIPLE. PREV% = 1 IMPLIES */ JM
012320 /* WE JUST FOUND A PAST PARTICIPLE ON OUR LAST PASS THROUGH AND */ JM
012330 /* 'TO BE' WAS FOUND EARLIER IN THE CLAUSE AND IF WE NOW FIND */ JM
012340 /* 'BY', FULL PASSIVE VOICE IS REALIZED (PASSV% = 7). OTHERWISE, */ JM
012350 /* MAKE SURE PREV% IS RESET. LIKEWISE, BYFLAG% = 1 IMPLIES WE */ JM
012360 /* HAVE JUST FOUND 'BY' AND WE'RE NOW LOOKING FOR 'TO BE' */ JM
012370 /* IMMEDIATELY FOLLOWING. IF 'TO BE' FOUND, SET PASSV% = 4 SO */ JM
012380 /* 'TO BE' CODE WILL INCREMENT PASSV% TO 6 AND ALL WE NEED IS */ JM
012390 /* PAST PARTICIPLE. */ JM
012400 JM
012410 /* 'TO BE' DOESN'T BEGIN SENTENCES, EXCEPT 'BE' IMPERATIVE */ JM
012420 JM
012430 IF SENTPOS$ = "B" AND WORD1$ <> "BE" THEN CHKDN JM
012440 IF PASSV% = "Y" AND STR(RECORD$, 1, 3) = "BY " AND PREV% = 1 ! JM
012450 THEN PASSV% = 7 JM
012460 IF PASSV% = "Y" AND STR(RECORD$, 24, 1) = "" AND BYFLAG% = 1 ! JM
012470 THEN PASSV% = 4 JM
012480 PREV%, BYFLAG% = 0 JM
012490 IF PASSV% = 3 THEN PASSV% = 0 /* 'BY' AFTER 1ST P. PARTICIPLE */ JM
012500 IF PASSV% <> "Y" THEN CHKTPV /* PASSV OPTION NOT ON */ JM
012510 IF (STR(RECORD$, 24, 1) = " " AND STR(RECORD$, 22, 1) = " ") THEN ! JM
012520 CHKBY /* NO PASSIVE OR TRUNCATED PASSIVE CHECK */ JM
012530 JM
012540 /* CHECK FOR PAST PARTICIPLE - ELSE DROP TO VERB TO BE */ JM
012550 IF STR(RECORD$, 22, 1) = "" THEN PSTPART JM

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 13:49 12/14/84 Page 23  
Input File is ANALYSIS in Library CRES@SRC on Volume CRES01

```

012560                                     /* VERB TO BE */ JM
012570 IF PASSV% = 0 OR PASSV% = 4 THEN PASSV% = PASSV% + 1 JM
012580 GOTO CHKTPV JM
012590 JM
012600 PSTPART:                                     /* PAST PARTICIPLE */ JM
012610 IF PASSV% = 1 OR PASSV% = 5 THEN PASSV% = PASSV% + 2 JM
012620 IF PASSV% = 3 THEN PREV% = 1 /* SET TO FIND 'BY' NEXT */ JM
012630 JM
012640 CHKBY:                                     /* CHECK FOR 'BY' */ JM
012650 IF STR(RECORD$,1,3) = "BY " THEN BYFLAG% = 1 ! JM
012660 JM
012670 CHKTPV: IF TRNK_PV$ <> "Y" THEN CHKDN /* TRNK_PV OPTION NOT ON */ JM
012680 PREV%, BYFLAG% = 0 JM
012690 IF (STR(RECORD$,24,1) = " " AND STR(RECORD$,22,1) = " ") THEN ! JM
012700 CHKDN /* NO NEED FOR TRU. PASSIVE CHECK */ JM
012710 JM
012720 /* PAST PARTICIPLE */ JM
012730 IF STR(RECORD$,22,1) = "" THEN PSTPART1 JM
012740                                     /* VERB TO BE */ JM
012750 IF TPASSV% = 0 THEN TPASSV% = TPASSV% + 1 JM
012760 GOTO CHKDN JM
012770 JM
012780 PSTPART1:                                     /* PAST PARTICIPLE */ JM
012790 IF TPASSV% = 1 THEN TPASSV% = TPASSV% + 2 JM
012800 JM
012810 CHKDN:                                     /* CHECK FOR A DOUBLE NEGATIVE */ JM
012820 IF NEG$ <> "Y" THEN CNTPREP /* DON'T CHECK DOUBLE NEG. */ JM
012830 IF STR(RECORD$,23,1) = "" THEN ! JM
012840 NEG%(1+NEG%(3)) = NEG%(1+NEG%(3)) + 1 JM
012850 JM
012860 CNTPREP:                                     /* COUNT PREPOSITIONS */ JM
012870 IF PREP_CNT$ <> "Y" OR STR(RECORD$,21,1) <> "P" ! JM
012880 THEN CHKPS JM
012890 PS% = PS% + 1 JM
012900 JM
012910 CHKPS:                                     /* CHECK FOR PERSONAL SENTENCES */ JM
012920 IF PERS_SENT$ <> "Y" THEN CNTNU JM
012930 IF STR(RECORD$,21,1) = "O" AND STR(RECORD$,1,1) = "Y" ! JM
012940 THEN PERSENT$ = "Y" /* THE WORD IS A DERIVATIVE OF 'YOU' */ JM
012950 IF SENTPOS$ = "B" AND STR(RECORD$,21,1) = "V" ! JM
012960 THEN PERSENT$ = "Y" /* IMPLIED PERSONAL SENT. */ JM
012970 IF WORD1$ = "I" THEN PERSENT$ = "Y" /* 1ST PERSON SINGULAR */ JM
012980 IF WORD1$ = "WE" THEN PERSENT$ = "Y" /* 1ST PERSON PLURAL */ JM
012990 IF WORD1$ = "MY" THEN PERSENT$ = "Y" /* 1ST PERSON POSSESSIVE */ JM
013000 IF WORD1$ = "OUR" THEN PERSENT$ = "Y" /* PLURAL POSSESSIVE */ JM
013010 JM
013120 JM
013130                                     /* SEARCH FOR 'WILL' OR 'SHALL' */ JM
013140 IF STR(RECORD$,1,5) = "WILL " OR STR(RECORD$,1,6) = "SHALL " ! JM
013150 THEN SETPUTR JM
013160 GOTO SETPRES JM
013220 JM
013230 SETPUTR:                                     /* SET FUTURE TENSE */ JM
013240 VRBT% = 3 JM
013250 GOTO CNTNU JM
013260 JM
013270 SETPRES:                                     /* SET PRESENT TENSE */ JM

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 13:49 12/14/84 Page 24  
Input File is ANALYSIS in Library CRES@SRC on Volume CRES01

```

013280 IF VRBT% = 0 THEN VRBT% = 1 JM
013290 JM
013300 CNTNU: IF PASSV$(<)"Y" THEN CDNF1/* CONTINUE CHECKING, SET FLAGS */ JM
013310 IF TRNK_PV$="Y" THEN CDNF1 /* ALL REGULAR PV'S ARE TRNK PV'S */ JM
013320 IF PASSV%(<) 7 THEN CDNF1 /* NOT YET */ JM
013330 PASSVF$(IW%) = "Y" /* SET FLAG */ JM
013340 PASSV% = 0 /* RESET VALUE */ JM
013350 JM
013360 CDNF1: IF TRNK_PV$(<)"Y" THEN CDNF /* TRUNCATED PASSIVE VOICE */ JM
013370 IF TPASSV%(<) 3 THEN CDNF /* NOT YET */ JM
013380 PASSVF$(IW%) = "Y" /* SET FLAG */ JM
013390 TPASSV%=0 /* RESET TRUNCATED PASSV FLAG */ JM
013400 JM
013410 CDNF: /* CHECK DOUBLE NEGATIVE FLAG */ JM
013420 IF NEG%(1)(<) 2 THEN NG /* NOT YET */ JM
013430 DNFL$(IW%) = "Y" /* SET FLAG */ JM
013440 NEG%(1) = 0 /* CLEAR VALUE */ JM
013450 GOTO CHKNOUN /* ONE AT A TIME */ JM
013470 JM
013480 NG: JM
013490 IF NEG%(2)(<) 2 THEN CHKNOUN /* NOT HERE EITHER */ JM
013500 DNFL$(IW%) = "Y" /* SET FLAG */ JM
013510 NEG%(2) = 0 /* CLEAR VALUE */ JM
013520 JM
013530 CHKNOUN: /* CHECK NOUN STRING COUNT */ JM
013540 IF NOUNSTR%(<) 4 THEN CHKLCLS /* NOT YET */ JM
013550 NOUNFL$(IW%) = "Y" /* SET FLAG */ JM
013560 NOUNSTR% = 0 /* CLEAR VALUE */ JM
013570 JM
013580 CHKLCLS: /* CHK END OF CLAUSE, END OF SENT. AND LONG CLAUSE */ JM
013590 IF CLSPOS$(<) "E" THEN CHKLSENT /* NOT END OF CLAUSE */ JM
013600 IF NC% > 10 THEN LCLSFL$(IW%) = NC% /* SET FLAG */ JM
013610 NC% = 0 JM
013620 JM
013630 CHKLSENT: /* CHECK LONG SENTENCES */ JM
013640 IF SENTPOS$(<) "E" THEN CW_RET JM
013650 IF WS% < 23 THEN CHKNPR JM
013660 LSENT$(IW%) = WS% JM
013670 LS% = LS% + 1 JM
013680 JM
013690 CHKNPR: /* CHECK FOR TOO MANY PREPS */ JM
013700 IF PREP_CNT$(<) "Y" OR PS% = 0 /* DON'T CHECK PREPS */ JM
013710 THEN CLEARFLDS JM
013720 J = WS% * 10 / PS% /* GET RATIO */ JM
013730 IF PS% > 2 AND J < 54 THEN PREPFL$(IW%) = PS% /* SET FLAG */ JM
013740 JM
013750 CLEARFLDS: /* CLEAR COUNT FIELDS */ JM
013760 WS%, PS% = 0 JM
013770 SP% = SP% + 1 /* COUNT SENT. / PAR. */ JM
013780 SC% = SC% + 1 /* COUNT SENTENCES */ JM
013790 EOS%=1 /* SET END OF SENTENCE FLAG */ JM
013860 IF PERSENT$(<) "Y" THEN CLRWRK /* NO PERS. SENT FOUND */ JM
013870 PE% = PE% + 1 /* COUNT PERS. SENT. */ JM
013880 PERSENT$ = " " JM
013890 JM
013900 CLRWRK: /* CLEAR WORK AREA */ JM
013910 MAT NEG% = ZER JM

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 13:49 12/14/84 Page 25  
Input File is ANALYSIS in Library CRES@SRC on Volume CRES01

```

013920 JM
013930 /* CHECK VERB AGREEMENT */ JM
013940 IF VRBAG% = 0 THEN VRBAG% = VRBT% JM
013950 IF VRBAG% <> VRBT% THEN VRBAG% = -99 JM
013960 VRBT% = 0 JM
013970 JM
013980 CW_RET: JM
013990 IF SENTPOS%="B" THEN SENTPOS%=" " /* OK HERE */ JM
014010 I% = 1 : RETURN /* CHECK WORD SUBROUTINE RETURN */ JM
014020 JM
014030 /****** JM
014040 /* */ JM
014050 /* SUBROUTINE UPDATE KEEPS THE USER ABREAST OF PROGRESS OF THE */ JM
014060 /* ANALYSIS BY DISPLAYING MESSAGES OF PROCEEDINGS AT THE */ JM
014070 /* TERMINAL. */ JM
014080 /* */ JM
014090 /****** JM
014100 JM
014110 UPDATE: WW1%=WW1%+WW% :WW%=0 /* KEEP TOTALS */ JM
014120 IF WW1%=0 THEN UPDMSG$="Working... Performing CRES Analysis..." JM
014130 IF WW1%=0 THEN LUPE /* SKIP COMPUTING A MESSAGE */ JM
014140 CONVERT WW1% TO WW1$,PIC(#####) /* CONVERT TO ALPHANUMERIC */ JM
014150 LN%=LEN(WW1$) /* COMPUTE THE LENGTH */ JM
014160 JM
014170 /* DECIDE WHICH MESSAGE TO DISPLAY TO THE USER */ JM
014180 JM
014190 IF STR(WW1$,LN%-2,3)="250" THEN UPDMSG$= STR(WW1$,1,LN%): JM
014200 & " Words Processed - ANALYSIS Proceeding..." JM
014210 IF STR(WW1$,LN%-2,3)="500" THEN UPDMSG$= WW1$ & " Words Processed : JM
014220 - ANALYSIS Continuing..." JM
014230 IF STR(WW1$,LN%-2,3)="750" THEN UPDMSG$= WW1$ & " Words Examined -! JM
014240 ANALYSIS Proceeding..." JM
014250 IF STR(WW1$,LN%-2,3)="000" THEN UPDMSG$= WW1$ & " Words Analyzed -! JM
014260 ANALYSIS Continuing..." JM
014270 LUPE: JM
014280 DISPLAY : JM
014290 AT (01,03), : JM
014300 "CRES: Document Analysis Subsystem (Status of Ana: JM
014310 lysis Screen)", : JM
014320 AT (02,03), : JM
014330 "===== : JM
014340 =====", : JM
014350 AT (12,20), : JM
014360 UPDMSG$,CH(51) JM
014370 RETURN /* UPDATE SUBROUTINE END */ JM
014380 JM
014390 JM
014400 /****** JM
014410 /* */ JM
014420 /* THIS SUBROUTINE COUNTS THE SYLLABLES IN A WORD. */ JM
014430 /* */ JM
014440 /****** JM
014450 JM
014460 SYLLY: JM
014470 JM
014480 /* CLEAR COUNT */ /* JM
014490 JM

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 13:49 12/14/84 Page 26  
Input File is ANALYSIS in Library CRES@SRC on Volume CRES01

```

014500      N%=0          /* USED TO COUNT SYLLABLES FROM THE FRONT */      JM
014510      R%=0          /* USED TO COUNT SYLLABLES FROM THE REAR */      JM
014520      M%=LEN%
014530      IF LEN%=1 THEN FINIS                                          JM
014540
014550      /* CHECK FOR A NUMBER IN THIS WORD */                          JM
014560
014570      Z99$=STR(WORD1$,1,LEN%)
014580      A0%=POS(Z99$="0")          /* LOOK FOR ZERO */                JM
014590      A1%=POS(Z99$="1")          /* LOOK FOR ONE */                 JM
014600      A2%=POS(Z99$="2")          /* LOOK FOR TWO */                 JM
014610      A3%=POS(Z99$="3")          /* LOOK FOR THREE */              JM
014620      A4%=POS(Z99$="4")          /* LOOK FOR FOUR */              JM
014630      A5%=POS(Z99$="5")          /* LOOK FOR FIVE */              JM
014640      A6%=POS(Z99$="6")          /* LOOK FOR SIX */                JM
014650      A7%=POS(Z99$="7")          /* LOOK FOR SEVEN */              JM
014660      A8%=POS(Z99$="8")          /* LOOK FOR EIGHT */             JM
014670      A9%=POS(Z99$="9")          /* LOOK FOR NINE */              JM
014680      A10%=A0%+A1%+A2%+A3%+A4%+A5%+A6%+A7%+A8%+A9%              JM
014690      IF A10%<>0 THEN CURVE      /* NUMBER FOUND; ONLY 1 SYLLABLE */ JM
014700
014710      /* SET START POINTER */                                       JM
014720
014730      K%=1
014740      IF LEN%<3 THEN VENT                                          JM
014750
014760      /* STRIP UNUSED ENDINGS */                                     JM
014770
014780      IF WORD1$="AREA" OR WORD1$="AREAS" THEN AREA                  JM
014790
014800      /* THESE NEXT TWO 'IF' STATEMENTS CATCH WORDS WITH UNUSUAL */ JM
014810      /* ENDINGS (LIKE 'RADIO', 'AUDIO', AND SOME PLURALS) AND AID */ JM
014820      /* THIS SUBROUTINE IN CORRECTLY COUNTING THEIR SYLLABLES. */ JM
014830
014840      IF POS("OA"=STR(WORD1$,LEN%,1))>0 AND STR(WORD1$,LEN%,1)<> : JM
014850      STR(WORD1$,LEN%-1,1) AND POS("AEIOUY"=STR(WORD1$,LEN%-2,1)) : JM
014860      =0 AND POS("AIOUY"=STR(WORD1$,LEN%-1,1))>0 THEN AREA          JM
014870      IF POS("OA"=STR(WORD1$,LEN%-1,1))>0 AND STR(WORD1$,LEN%-1,1) : JM
014880      <>STR(WORD1$,LEN%-2,1) AND POS("AEIOUY"=STR(WORD1$,LEN%-3,1)) : JM
014890      =0 AND POS("AIOUY"=STR(WORD1$,LEN%-2,1))>0 AND                JM
014900      STR(WORD1$,LEN%,1)="S" THEN AREA                                JM
014910
014920      /* THIS NEXT BLOCK OF CODE STRIPS OFF FRONT AND TRAILING */ JM
014930      /* PARTS OF SPECIFIC WORDS, AND HELPS US HANDLE THINGS LIKE */ JM
014940      /* IMBEDDED SILENT 'E'S IN WORDS. */                            JM
014950
014960      VENT:IF STR(WORD1$,1,5)="THERE" AND M%>5 THEN THES            JM
014970      IF STR(WORD1$,LEN%-3,4)="NING" THEN NING                        JM
014980      IF STR(WORD1$,LEN%-3,4)="IALS" THEN MENT                         JM
014990      IF STR(WORD1$,LEN%-2,3)="DII" THEN AREA                         JM
015000      IF STR(WORD1$,LEN%-2,3)="ING" THEN THREE                       JM
015010      IF STR(WORD1$,LEN%-2,3)="IER" THEN TWO                         JM
015020      IF STR(WORD1$,LEN%-2,3)="IOR" THEN THREE                       JM
015030      IF STR(WORD1$,LEN%-2,3)="IAL" THEN CHKD                        JM
015040      IF STR(WORD1$,LEN%-2,3)="QUE" THEN QUE                         JM
015050      IF STR(WORD1$,LEN%-2,3)="GUE" THEN QUE                         JM
015060      IF STR(WORD1$,LEN%-1,2)="AL" THEN TWO                          JM

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 13:49 12/14/84 Page 27  
Input File is ANALYSIS in Library CRES@SRC on Volume CRES01

```

015070 IF STR(WORD1$,LEN%-1,2)="LY" THEN TWO JM
015080 IF STR(WORD1$,LEN%-1,2)="TY" THEN TWO JM
015090 IF STR(WORD1$,LEN%-3,4)="YARD" THEN MENT JM
015100 IF STR(WORD1$,LEN%-3,4)="MENT" THEN MENT JM
015110 IF STR(WORD1$,LEN%-4,5)="MENTS" THEN MENTS JM
015120 IF STR(WORD1$,LEN%,1)="D" OR STR(WORD1$,LEN%,1)="S" THEN EDES JM
015130 IF STR(WORD1$,LEN%,1)="E" THEN FINALE JM
015140 GOTO VOWEL JM
015150 JM
015160 QUE: M%=M%-3 /* STRIP "QUE" OR "GUE" FROM END (NO SYLLABLES!) */ JM
015170 GOTO VOWEL JM
015180 JM
015190 AREA: IF STR(WORD1$,LEN%,1)<>"S" THEN M%=M%-3 ! JM
015200 ELSE M%=M%-4 /* HANDLES AUDIO, AREA, RADIO, ETC. */ JM
015210 R%=R%+2 JM
015220 IF M%>1 THEN VOWEL JM
015230 GOTO FINIS JM
015240 JM
015250 /* CHECK FOR CERTAIN CONSONANTS PRECEEDING "IAL" */ JM
015260 JM
015270 CHKD: Z1$=STR(WORD1$,LEN%-3,1) JM
015280 IF Z1$="D" OR Z1$="N" OR Z1$="R" OR Z1$="V" THEN TWO JM
015290 GOTO THREE JM
015300 JM
015310 THES: R%=R%+1 /* ADD 1 TO SYLLABLE COUNT */ JM
015320 K%=6 JM
015330 IF M%<K% THEN FINIS JM
015340 GOTO VOWEL JM
015350 JM
015360 TWO: R%=R%+1 /* ADD 1 TO SYLLABLE COUNT */ JM
015370 M%=M%-2 /* DON'T LOOK AT THE LAST 2 LETTERS */ JM
015380 IF M%>1 THEN VOWEL JM
015390 GOTO FINIS JM
015400 JM
015410 THREE: R%=R%+1 /* ADD 1 TO SYLLABLE COUNT */ JM
015420 M%=M%-3 /* DON'T LOOK AT THE LAST 3 LETTERS */ JM
015430 IF M%>1 THEN VOWEL JM
015440 GOTO FINIS JM
015450 JM
015460 NING: IF STR(WORD1$,LEN%-5,3) = "ION" THEN THREE JM
015470 MENT: R%=R%+1 /* ADD 1 TO SYLLABLE COUNT */ JM
015480 M%=M%-4 /* DON'T LOOK AT THE LAST 4 LETTERS */ JM
015490 IF M%>1 THEN VOWEL JM
015500 GOTO FINIS JM
015510 JM
015520 JM
015530 MENTS: R%=R%+1 /* ADD 1 TO SYLLABLE COUNT */ JM
015540 M%=M%-5 /* DON'T LOOK AT THE LAST 5 LETTERS */ JM
015550 IF M%>1 THEN VOWEL JM
015560 GOTO FINIS JM
015570 JM
015580 /* CHECK FOR ED OR ES */ JM
015590 JM
015600 EDES: IF STR(WORD1$,LEN%-1,1)<>"E" AND STR(WORD1$,LEN%-1,1)<>"H" THEN TWO JM
015610 7) THEN VOWEL JM
015620 IF STR(WORD1$,LEN%-2,3)="TED" OR STR(WORD1$,LEN%-2,3)="DED" ! JM
015630 OR STR(WORD1$,LEN%-2,3)="CES" THEN R%=R%+1 /*ADD A SYL.! */ JM

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 13:49 12/14/84 Page 28  
Input File is ANALYSIS in Library CRES@SRC on Volume CRES01

```

015640 IF STR(WORD1$,LEN%-2,3)="SES" THEN R%=R%+1 /* ADD A SYL.! */ JM
015650 IF M%<4 OR R%>0 THEN SKEP /* CHECK FOR COMBOS BEFORE 'ES' */ JM
015660 IF STR(WORD1$,M%-1,2)<>"ES" THEN SKEP /* DITTO */ JM
015670 Z$=STR(WORD1$,M%-3,2) /* SEE IF COMBOS. PRECEED "ES" */ JM
015680 A1%=POS(Z$="A") JM
015690 A2%=POS(Z$="I") JM
015700 A3%=POS(Z$="U") JM
015710 A4%=POS(Z$="O") JM
015720 A7%=A1%+A2%+A3%+A4% JM
015730 IF A7%<>0 THEN SKEP /* VOWEL - NO MORE SYLLABLES */ JM
015740 GOTO VOWEL /* A CONS. - ADD A SYLLABLE */ JM
015750 SKEP: IF STR(WORD1$,M%-1,2)<>"ED" THEN SKEP1 /* DITTO */ JM
015760 IF M%<5 OR R%>0 THEN SKEP1 /* CHECK FOR 3 CONS. BEFORE 'ED' */ JM
015770 Z2$=STR(WORD1$,M%-4,3) /* SEE IF 3 CONS. PRECEED "ED" */ JM
015780 A1%=POS(Z2$="A") JM
015790 A2%=POS(Z2$="E") JM
015800 A3%=POS(Z2$="I") JM
015810 A4%=POS(Z2$="O") JM
015820 A5%=POS(Z2$="U") JM
015830 A6%=POS(Z2$="Y") JM
015840 A7%=A1%+A2%+A3%+A4%+A5%+A6% JM
015850 IF A7%<>0 THEN SKEP1 /* VOWEL - NO MORE SYLLABLES */ JM
015860 R% = R%+1 /* A CONS. - ADD A SYLLABLE */ JM
015870 SKEP1: M% = M%-2 JM
015880 IF M%>1 THEN VOWEL JM
015890 GOTO FINIS JM
015900 JM
015910 /* CHECK FOR FINAL E, NOT LE */ JM
015920 JM
015930 FINALE: IF STR(WORD1$,LEN%-1,1) = "L" THEN VOWEL JM
015940 M% = M% -1 JM
015950 IF M%>1 THEN VOWEL JM
015960 GOTO FINIS JM
015970 JM
015980 /* CHECK FOR TRAILING 'Y' */ JM
015990 JM
016000 TRAILY: IF STR(WORD1$,K%,1)="Y" THEN CNTSYL JM
016010 JM
016020 /* CHECK FOR VOWEL */ JM
016030 JM
016040 VOWEL: IF STR(WORD1$,K%,1)="A" OR STR(WORD1$,K%,1)="U" THEN ONEXT JM
016050 IF STR(WORD1$,K%,1)="E" THEN INEXT JM
016060 IF STR(WORD1$,K%,1)="I" THEN U_ION JM
016070 IF STR(WORD1$,K%,1)="O" THEN EORA JM
016080 GOTO GETNXT JM
016090 JM
016100 /* CHECK FOR O NEXT */ JM
016110 JM
016120 ONEXT: IF K%=M% THEN SKIPV JM
016130 IF STR(WORD1$,K%+1,1) = "O" THEN CNTSYL JM
016140 IF STR(WORD1$,K%+1,1) = "A" THEN CHKUA JM
016150 GOTO SKIPV JM
016160 JM
016170 /* CHECK FOR "UA" WITH A "G" OR "Q" PRECEEDING */ JM
016180 JM
016190 CHKUA: IF STR(WORD1$,K%,2)<>"UA" OR K%<2 THEN CNTSYL JM
016200 IF STR(WORD1$,K%-1,1)="G" OR STR(WORD1$,K%-1,1)="Q" THEN : JM

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 13:49 12/14/84 Page 29  
Input File is ANALYSIS in Library CRES@SRC on Volume CRES01

```

016210      SKIPV                                     JM
016220      GOTO CNTSYL                                JM
016230                                     JM
016240      /* CHECK FOR O NEXT */                    JM
016250                                     JM
016260 ONEX:  IF K% = M% THEN SKIPV                    JM
016270          IF STR(WORD1$,K%+1,1) = "O" THEN CNTSYL JM
016280          GOTO SKIPV                                JM
016290                                     JM
016300      /* CHECK FOR I NEXT */                    JM
016310                                     JM
016320 INEXT: IF K% = M% THEN CHKCONS                  /* CHECK FOR PRECEEDING CONS. */ JM
016330          IF K% = 2 AND M% > 5 THEN DEAL          /* SEE IF RE OR DE IN FRONT */ JM
016340 GETBAK: IF STR(WORD1$,K%+1,1) = "I" THEN SKIPV  /* CNTSYL */ JM
016350          GOTO ONEX                                    JM
016360                                     JM
016370 /* CODE TO DEAL WITH LEADING RE & DE */          JM
016380                                     JM
016390 DEAL:  IF STR(WORD1$,1,1) <> "D" AND STR(WORD1$,1,1) <> "R" THEN GETBAK JM
016400          Z1$ = STR(WORD1$,K%+1,1)                JM
016410          A1$ = POS(Z1$,"A")                        JM
016420          A2$ = POS(Z1$,"E")                        JM
016430          A3$ = POS(Z1$,"I")                        JM
016440          A4$ = POS(Z1$,"O")                        JM
016450          A5$ = POS(Z1$,"U")                        JM
016460          A7$ = A1$ + A2$ + A3$ + A4$ + A5$          JM
016470          IF A7$ = 0 THEN GETBAK                    /* NO VOWEL - NOT 2 SYL. ! */ JM
016480          N% = N% + 1                                /* ADD A SYLLABLE */ JM
016490          K% = K% + 1                                /* INCREMENT PAST "RE", "DE", ETC. */ JM
016500          GOTO VOWEL                                /* TRY AGAIN */ JM
016510                                     JM
016520      /* CHECK FOR U/ION */                      JM
016530                                     JM
016540 U_ION: IF K% = M% THEN SKIPV                    JM
016550          IF STR(WORD1$,K%+1,1) = "U" THEN CNTSYL  JM
016560          IF K% + 2 > M% THEN ONEXT                JM
016570          IF STR(WORD1$,K%,2) = "IA" THEN IA        JM
016580 BACK:  IF STR(WORD1$,K%+1,1) <> "O" THEN SKIPV  JM
016590          IF STR(WORD1$,K%+2,1) <> "N" THEN CNTSYL JM
016600          GOTO SKIPV                                JM
016610                                     JM
016620      /* CHECK FOR E OR A */                      JM
016630                                     JM
016640 EORA:  IF K% = M% THEN SKIPV                    JM
016650          IF STR(WORD1$,K%+1,1) = "E" THEN CNTSYL  JM
016660          GOTO SKIPV                                JM
016670                                     JM
016680      /* GET NEXT CHARACTER */                    JM
016690                                     JM
016700 GETNXT: K% = K% + 1                                JM
016710          IF K% > M% THEN FINIS                      JM
016720          GOTO TRAILY                                JM
016730                                     JM
016740      /* COUNT SYLLABLES */                      JM
016750                                     JM
016760 CNTSYL: N% = N% + 1                                JM
016770          GOTO GETNXT                                JM

```

AD-A168 681

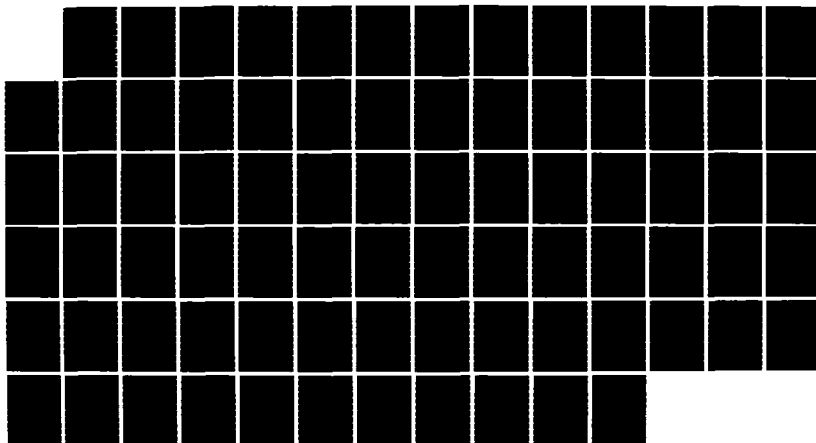
COMPUTER READABILITY AND EDITING SYSTEM (CRES)  
PROGRAMMER'S REFERENCE MANUAL(U) NAVAL TRAINING SYSTEMS  
CENTER ORLANDO FL S R SCHALOW ET AL. JAN 86  
NAVTRASYSCEN-TR-86-001

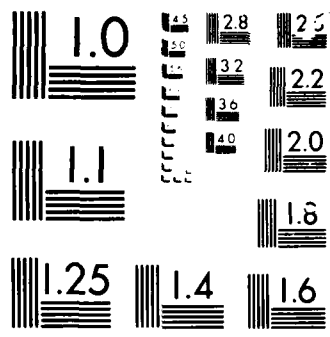
4/4

UNCLASSIFIED

F/G 5/9

NL





MICROCOPY

CHART

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 13:49 12/14/84 Page 30  
Input File is ANALYSIS in Library CRES@SRC on Volume CRES01

```

016780 JM
016790 /* SKIP A VOWEL */ JM
016800 JM
016810 SKIPV: K% = K% + 1 JM
016820 GOTO CNTSYL JM
016830 JM
016840 IA: IF K% < 2 THEN BACK /* NO NEED TO CHECK */ JM
016850 IF STR(WORD1$, K% - 1, 1) = "D" THEN CNTSYL JM
016860 IF STR(WORD1$, K% - 1, 1) = "C" THEN CNTSYL JM
016870 IF STR(WORD1$, K% - 1, 1) = "R" THEN CNTSYL JM
016880 IF STR(WORD1$, K% - 1, 1) = "V" THEN CNTSYL JM
016890 IF K% < 3 THEN BACK /* NO NEED TO CHECK */ JM
016900 Z$ = STR(WORD1$, K% - 2, 2) /* SEE IF 2 CONS. PRECEED "IA" */ JM
016910 A1% = POS(Z$ = "A") JM
016920 A2% = POS(Z$ = "E") JM
016930 A3% = POS(Z$ = "I") JM
016940 A4% = POS(Z$ = "O") JM
016950 A5% = POS(Z$ = "U") JM
016960 A6% = POS(Z$ = "Y") JM
016970 A7% = A1% + A2% + A3% + A4% + A5% + A6% JM
016980 IF A7% < 0 THEN BACK /* VOWEL - NO MORE SYLLABLES */ JM
016990 GOTO CNTSYL /* 2 CONS. - ADD A SYLLABLE */ JM
017000 JM
017010 CHKCONS: IF K% < 3 THEN FINIS /* NO NEED TO CHECK */ JM
017020 Z$ = STR(WORD1$, K% - 2, 2) /* SEE IF 2 CONS. PRECEED "E" */ JM
017030 A1% = POS(Z$ = "A") JM
017040 A2% = POS(Z$ = "E") JM
017050 A3% = POS(Z$ = "I") JM
017060 A4% = POS(Z$ = "O") JM
017070 A5% = POS(Z$ = "U") JM
017080 A6% = POS(Z$ = "Y") JM
017090 A7% = A1% + A2% + A3% + A4% + A5% + A6% JM
017100 IF A7% < 0 THEN FINIS /* VOWEL - NO MORE SYLLABLES */ JM
017110 N% = N% + 1 /* 2 CONS. - ADD A SYLLABLE */ JM
017120 JM
017130 /* EVEN A NULL WORD GETS 1 SYLLABLE */ JM
017140 JM
017150 FINIS: IF N% > 0 OR K% < 1 THEN CURVE JM
017160 Z3$ = STR(WORD1$, 1, K%) JM
017170 A1% = POS(Z3$ = "A") JM
017180 A2% = POS(Z3$ = "E") JM
017190 A3% = POS(Z3$ = "I") JM
017200 A4% = POS(Z3$ = "O") JM
017210 A5% = POS(Z3$ = "U") JM
017220 A6% = POS(Z3$ = "Y") JM
017230 A7% = A1% + A2% + A3% + A4% + A5% + A6% JM
017240 IF A7% = 0 THEN CURVE /* VOWEL - NO MORE SYLLABLES */ JM
017250 N% = 1 JM
017260 CURVE: TLC% = N% + R% JM
017270 TLC% = MAX(TLC%, 1) JM
017280 RETURN /* SYLLY SUBROUTINE RETURN */ JM
017290 JM
017300 /****** JM
017310 /* JM
017320 /* THIS SUBROUTINE ADDS A WORD TO THE UNCOMMON WORD LIST. JM
017330 /* JM
017340 /****** JM

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 13:49 12/14/84 Page 31  
Input File is ANALYSIS in Library CRES@SRC on Volume CRES01

```

017350 JM
017360 ADD_UNCWL: JM
017370   UNCPT%=MIN(21,UNCPT%) JM
017380   FOR V% = 1 TO UNCPT%-1 JM
017390     FOUND% = 0 /* RESET FOUND FLAG */ JM
017400     FOR S% = 1 TO IDX% /* LOOP TO FIND WORD IN ARRAY */ JM
017410       IF UNC$(V%) > UCW$(S%) THEN LOOP JM
017420       IF UNC$(V%) = UCW$(S%) THEN INCREM /* JUST INC. COUNT */ JM
017430 JM
017440       /* WORD SHOULD BE INSERTED IN THIS SPOT */ JM
017450 JM
017460       JJ%=IDX%+1 JM
017470       FOR L%=JJ% TO S%+1 STEP -1 JM
017480         UCW$(L%)=UCW$(L%-1) /* MOVE WORDS DOWN ONE SLOT */ JM
017490         UCWCT%(L%)=UCWCT%(L%-1) /* MOVE COUNT ALSO */ JM
017500       NEXT L% JM
017510       UCW$(S%)=UNC$(V%) /* PLACE THE WORD IN THE ARRAY */ JM
017520       UCWCT%(S%)=1 /* PLACE THE COUNT IN THE ARRAY */ JM
017530       S%=IDX% /* GETS US OUT OF THE MIDDLE LOOP */ JM
017540       IDX%=JJ% /* INCREMENT THE INDEX TO THE ARRAY */ JM
017550       FOUND%=1 /* SET THE FOUND FLAG */ JM
017560       GOTO LOOP JM
017570 INCREM: UCWCT%(S%)=UCWCT%(S%)+1 JM
017580       FOUND%=1 JM
017590       S%=IDX% /* GETS US OUT OF THE BIG LOOP */ JM
017600 LOOP: NEXT S% JM
017610   IF FOUND%=1 THEN RETO JM
017620   IDX%=IDX%+1 /* INCREMENT THE INDEX TO THE ARRAY */ JM
017630   UCW$(IDX%)=UNC$(V%) /* PUT THE WORD AT THE END */ JM
017640   UCWCT%(IDX%)=1 /* SET THE COUNT TO ONE */ JM
017650 RETO: NEXT V% JM
017660   UNC$()=HEX(20) /* RESET BUFFER VARIABLES */ JM
017670   UNCPT%=1 JM
017680 RETURN /* ADD_UNCWL SUBROUTINE RETURN */ JM
017690 JM
017700 /****** JM
017710 /* JM
017720 /* THIS SUBROUTINE CHECKS FOR POSSIBLE PHRASES. JM
017730 /* JM
017740 /****** JM
017760 CHKPHSPL: JM
017780   I%=1 JM
017790   K%=0 JM
017800   IF CW$="P" THEN ADD_NWD /* PARTIAL PHRASE */ JM
017810   IF CW$="X" THEN RD /* RETRY */ JM
017820   IF SENTPO$="E" OR CLSPO$="E" THEN CHKNWDS /*NO POSS. PH. */ JM
017840   IP%=1 JM
017850   WORDP$=ALL(HEX(20)) /* BLANK OUT OUR PHRASE BUFFER */ JM
017860 SAVPH: STR(WORDP$,IP%,LEN%)=WORD1$ JM
017870   IP%=IP%+LEN% JM
017880 RD: READ #4, KEY> = STR(WORDP$,1,IP%),PREC$,EOD GOTO CHKNUM2 JM
017890   IF STR(WORDP$,1,IP%) <> STR(PREC$,1,IP%) THEN CHKNUM2 JM
017900 JM
017910   /* WE MUST HAVE A POSSIBLE PARTIAL PHRASE */ JM
017920 JM
017930   CW$="P" JM
017940   GOTO PHRS_RET JM

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 13:49 12/14/84 Page 32  
Input File is ANALYSIS in Library CRES@SRC on Volume CRES01

```

017950 ADD_NWD: JM
017960 IF (IP%+LEN%)>23 THEN SWITCH JM
017970 STR(WORDP$,IP%+1,LEN%)=WORD1$ JM
017980 IP%=IP%+LEN%+1 JM
017990 READ #4, KEY)= STR(WORDP$,1,IP%),PREC$,EOD GOTO CHOP JM
018000 JM
018010 /* LET'S SEE IF THE FIRST PART MATCHES */ JM
018020 JM
018030 IF STR(WORDP$,1,IP%)<> STR(PREC$,1,IP%) THEN CHOP JM
018040 JM
018050 /* LET'S SEE IF THE WHOLE PHRASE MATCHES */ JM
018060 JM
018070 IF STR(WORDP$,1,24)=STR(PREC$,1,24) THEN SUCCESS JM
018080 JM
018090 /* DO WE KEEP LOOKING? */ JM
018100 JM
018110 IF IW% = 5 THEN CHOP JM
018120 IF SENTPOS$ = "E" OR CLSPOS$="E" THEN CHKNWDS JM
018130 GOTO PHRS_RET JM
018140 SUCCESS: JM
018150 CW$="Y" /* WE FOUND ONE - SET FLAG */ JM
018160 GOTO PHRS_RET JM
018170 JM
018180 CHKNWDS: JM
018190 IF IW% < 3 THEN FAIL JM
018200 GOTO SWITCH JM
018210 JM
018220 CHKNUM2: JM
018230 IF IW% = 1 THEN FAIL /* IF HERE, NO MATCH */ JM
018240 GOTO CHOP JM
018250 JM
018260 SWITCH: K% = -1 JM
018270 JM
018280 CHOP: CW$ = "X" /* CHOP OFF LEADING WORD - X MEANS RETRY */ JM
018290 I% = 1 JM
018300 D% = 3 /* PRINT WORDS AND FLAGS ONLY */ JM
018310 GOSUB PRT_WDS JM
018320 IF I% = 0 THEN PHRS_RET JM
018330 J% = POS(STR(WORDP$,1,24) = " ") /* FIND THE FIRST BLANK */ JM
018340 WORDP$ = STR(WORDP$,J%+1) /* CHOP OFF 1ST WORD */ JM
018350 IP% = IP% - J% /*: IW% = IW% - 1 */ /* ADJUST BUFFER PTR. */ JM
018360 IF K% = 0 THEN RD JM
018370 GOTO SAVPH JM
018380 FAIL: CW$ = "N" /* NO PHRASE - SET FLAG */ JM
018390 PHRS_RET: JM
018400 RETURN /* CHKPHSFL SUBROUTINE RETURN */ JM
018410 JM
018420 /***** JM
018430 /* JM
018440 /* THIS SUBROUTINE SELECTS A LINE TO PRINT. JM
018450 /* JM
018460 /***** JM
018470 JM
018480 SELOPT: JM
018490 JM
018500 D% = 1 /* ASSUME SUCCESS */ JM
018510 /* SELECT THE WAY TO GO */ JM

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 13:49 12/14/84 Page 33  
Input File is ANALYSIS in Library CRES@SRC on Volume CRES01

```

018520                                     JM
018530      IF CW$="Y" THEN FORTY /* PHRASE */      JM
018540      IF WSL$="Y" THEN THIRTY8 /* WORD SUBSTITUTES */      JM
018550                                     JM
018560 /* IF NONE OF THE ABOVE, PRINT WORDS */      JM
018570                                     JM
018580      GOSUB PRNT_WDS      JM
018590      GOTO OUTS      JM
018600                                     JM
018610 /* SET UP DATA SO WORD SUBS. CAN BE FETCHED USING PHRASE CODE */      JM
018620                                     JM
018630 THIRTY8: IF PHCT%=1 THEN THIRTY9 /* DON'T FORGET ANY WORDS */      JM
018640                                     JM
018650 /* INCLUDE WORDS SAVED FOR POSSIBLE PHRASE BEFORE */      JM
018660 /* IS THERE ROOM ON THE OUTPUT LINE? */      JM
018670                                     JM
018680      FOR I%=1 TO PHCT%-1      JM
018690      IF IO% + SAVPHL%(I%) + SAVSUBL%(I%) > LL% THEN GOSUB PRTLINE      JM
018700      IF PFLAG%=0 THEN OUTS      JM
018710      IF SAVSUBL%(I%)=0 THEN      JM
018720      STR(OLINE$,IO%,SAVPHL%(I%))=STR(SAVPH$(I%),1,SAVPHL%(I%))!      JM
018730      ELSE STR(OLINE$,IO%,SAVPHL%(I%)+SAVSUBL%(I%)) = "<" " !      JM
018740      STR(SAVPH$(I%),1,SAVPHL%(I%)) & ">" & !      JM
018750      STR(SAVSUB$(I%),1,SAVSUBL%(I%)-2)      JM
018760      WCL%=WCL%+1 /* INCREMENT THE WORD COUNT */      JM
018770      WPL%=WPL%+1 /* INCREMENT THE WORDS PER PARAGRAPH COUNT */      JM
018780      SLCL%=SLCL%+SAVSYL%(I%) /* INCREMENT SYLLABLE COUNT */      JM
018810      IO%=SAVPHL%(I%) + IO% + 1 + SAVSUBL%(I%)      JM
018820      IF SAVSUBL%(I%)<>0 THEN SAVSUBL%(I%)=0      JM
018830      NEXT I%      JM
018840      PHCT% = 1      JM
018850 THIRTY9: PREC$=ALL(HEX(20)) /* SET UP PREC$ WITH WORD SUBS. */      JM
018860      STR(PREC$,1,20)=STR(RECL$,1,20)      JM
018870      STR(PREC$,25)=STR(RECL$,21,20)      JM
018880      STR(PREC$,45)=STR(RECL$,41,15)      JM
018890      WSL$="N" /* RESET THE FLAG */      JM
018900                                     JM
018910 /* PRINT A PHRASE AND SUBSTITUTES & SET DEFAULTS */      JM
018920                                     JM
018930 FORTY: IF CW$="Y" THEN MAT SAVSUBL% = ZER /* CLEAR WORD SUBS. */      JM
018940      J%=0      JM
018950      K1%=0      JM
018960      K2%=0      JM
018970      K3%=1      JM
018980                                     JM
018990 /* COMPUTE THE LENGTH OF PHRASE AND SUBSTITUTES */      JM
019000 /* FIRST, GET THE LENGTH OF PHRASE */      JM
019010                                     JM
019020      IF PHCT%=1 THEN SKPTHS      JM
019030      FOR I%=1 TO PHCT% -1      JM
019040          J%=J%+SAVPHL%(I%)+1 /* INCLUDE SPACES */      JM
019050      NEXT I%      JM
019060 SKPTHS: IF STR(PREC$,25,1)=" " THEN BLANK$="Y" /*NO SUBS. */      JM
019070      IF BLANK$="Y" THEN J%=J%+WLEN% /* DON'T FLAG THE PHRASE */!      JM
019080      ELSE J%=J%+2+WLEN% /* INCLUDES LAST WORD AND <...> */      JM
019090                                     JM
019100 /* CHECK ON NUMBER OF SUBSTITUTES TO PRINT */      JM

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 13:49 12/14/84 Page 34  
Input File is ANALYSIS in Library CRES@SRC on Volume CRES01

```

019110                                     JM
019120     IF NUM_SUB$="0" OR STR(PREC$,25,1)=" " THEN FORTY2 /*NONE */ JM
019130                                     JM
019140     /* GET LENGTH OF SUBSTITUTE #1 */ JM
019150                                     JM
019160     K1=-1 JM
019170     K1=POS(STR(PREC$,25,20)=" ") JM
019180 OVA: IF STR(PREC$,K1+25,1)=" " OR K1=20 THEN COUNT JM
019190     K4=POS(STR(PREC$,K1+25,20-K1)=" ") /* GET THE REST */ JM
019200     IF K4=0 THEN K1=20 /*THE SUBSTITUTE TOOK 20 SPACES*/ JM
019210     K1=K1+K4 JM
019220     GOTO OVA JM
019230 COUNT: K1=MIN(20,K1) JM
019240     J=J+K1+5 /* INCLUDES "< "....">" */ JM
019250                                     JM
019260     /* ANOTHER SUBSTITUTE ? */ JM
019270                                     JM
019280     IF STR(PREC$,45,1)=" " THEN FORTY2 JM
019290     K2=-1 JM
019300     K2=POS(STR(PREC$,45,20)=" ") JM
019310 OVAL: IF STR(PREC$,K2+45,1)=" " OR K2=20 THEN COUNT1 JM
019320     K4=POS(STR(PREC$,K2+45,20-K2)=" ") /* GET THE REST */ JM
019330     IF K4=0 THEN K2=20 /* THE SUBSTITUTE TOOK 20 SPACES */ JM
019340     K2=K2+K4 JM
019350     GOTO OVAL JM
019360 COUNT1: K2=MIN(20,K2) JM
019380                                     JM
019390     J=J+K2+3 /* INCLUDES "/" & TWO SPACES */ JM
019400                                     JM
019410     /* PRINT THE PHRASE */ JM
019420                                     JM
019430 FORTY2: PFLAG= K3 JM
019440     IF IO+J > LL THEN GOSUB PRTLNE JM
019450     IF PFLAG=0 THEN OUTS JM
019460     IF BLANK$="Y" THEN MISS JM
019470     STR(OLINE$,IO,1)= "< " JM
019480     IO=IO+1 JM
019490                                     JM
019500     /* INSERT THE PHRASE */ JM
019510                                     JM
019520 MISS: SAVPH$(PHCT)=WORDS$ JM
019530     SAVPHL$(PHCT)=WLEN$ JM
019540     SAVSYL$(PHCT)=TLC$ JM
019550     FOR I=1 TO PHCT JM
019560         STR(OLINE$,IO,SAVPHL$(I))=STR(SAVPH$(I),1,SAVPHL$(I)) JM
019570         WCL=WCL+1 /* INCREMENT THE WORD COUNT */ JM
019580         WPL=WPL+1 /* INCREMENT THE WORDS PER PAR. COUNT */ JM
019590         SLC=SLC+SAVSYL$(I) /* INCREMENT THE SYLLABLE COUNT */ JM
019620         IO=IO+SAVPHL$(I)+1 /* INCLUDE SPACES */ JM
019630     NEXT I JM
019640     IF BLANK$(">")="Y" THEN IO=IO-1 /* EXCEPT THE LAST ONE */ JM
019650     PHCT=1 /* RESET PARTIAL PHRASE COUNT */ JM
019660                                     JM
019670     /* INSERT SUBSTITUTES */ JM
019680                                     JM
019690     IF BLANK$="Y" THEN FIFTY JM
019700     STR(OLINE$,IO,1)=">" JM

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 13:49 12/14/84 Page 35  
Input File is ANALYSIS in Library CRES@SRC on Volume CRES01

```

019710      IO%=IO%+1                      JM
019720      IF NUM_SUB$="0" THEN FIFTY      /* NO SUBSTITUTES */ JM
019730      STR(OLINE$,IO%,2)=""<"*      JM
019740      IO%=IO%+3                      /* INCLUDE A SPACE */ JM
019750      IF K1%=0 THEN FORTY4 /* NO SUBST. IN PHRASE SUB. LIST */ JM
019760      STR(OLINE$,IO%,K1%)=STR(PREC$,25,K1%) JM
019770      IO%=IO%+K1%                    JM
019780      IF K2%=0 THEN FORTY4            JM
019790      STR(OLINE$,IO%,1)=""/*          JM
019800      IO%=IO%+2                      /* INCLUDE A SPACE */ JM
019810      STR(OLINE$,IO%,K2%)=STR(PREC$,45,K2%) JM
019820      IO%=IO%+K2%                    JM
019830      JM
019840      /* FINISH THE ENTRY */          JM
019850      JM
019860      FORTY4: STR(OLINE$,IO%,2)=""*   JM
019870      IO%=IO%+2                      JM
019880      JM
019890      /* NOW PRINT ANY FLAGS FOR THE WORDS IN THE PHRASE */ JM
019900      JM
019910      FIFTY: BLANK$="N"              JM
019920      D%=2                          JM
019930      GOSUB PRNT_WDS                  JM
019940      JM
019950      OUTS: RETURN                    /* SELOPT SUB. RETURN */ JM
019960      JM
019970      /****** JM
019980      /* JM
019990      /* THIS SUBROUTINE PRINTS A LINE TO THE PRINT FILE. JM
020000      /* JM
020010      /****** JM
020020      JM
020030      PRTLINE: JM
020040      JM
020050      IF E_O_F$="E" THEN FINISH JM
020060      JM
020070      IF LC%=0 AND PC%=1 THEN GOSUB TOP1 JM
020080      C$ = HEX(0001) JM
020090      WRITE #12, USING Y, C$ JM
020100      WRITE #12, USING X, C$, OLINE$ JM
020110      OLINE$=ALL(HEX(20)) JM
020120      IO% = 1 JM
020130      LC% = LC% + 2 JM
020140      IF LC%<=LP% THEN FINAL /* NO NEED FOR RGL YET */ JM
020150      LC%=0 /* RESET LINE COUNT */ JM
020160      IF PC%<=GP% THEN TEN /* IS IT TIME FOR INTERIM ANALYSIS? */ JM
020170      GP% = GP% + BP% /* RESET GP% FOR NEXT BREAK POINT */ JM
020180      IF INT_ANL$<>"Y" THEN CHKUCW JM
020190      GOSUB PRTRGL /* PRINT INTERIM ANALYSIS */ JM
020200      GOTO CHKUCW JM
020210      FINISH: IF RGL$<>"Y" THEN CHKUCW /* SHOULD WE PRINT RGL? */ JM
020220      GOSUB PRTRGL JM
020230      CHKUCW: JM
020240      IF IDX% = 0 OR UNCOM$<>"Y" THEN TEN /* NO UNCOMMON WORDS */ JM
020250      GOSUB UWL /* PRINT UNCOMMON WORD LIST */ JM
020260      IF E_O_F$="E" THEN FINAL JM
020270      TEN: GOSUB TOP1 /* PRINT TOP OF PAGE */ JM

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 13:49 12/14/84 Page 36  
Input File is ANALYSIS in Library CRES@SRC on Volume CRES01

```

020280      GOTO FINAL                      JM
020290 TOP1: GOSUB TOP                      JM
020300      WRITE #12,USING ZZZ,C$,PC%      JM
020310      PC%=PC%+1                      JM
020320      RETURN                          JM
020330 X:### #####! JM
020340 ##### JM
020350 Y:### JM
020360 ZZZ:### ! JM
020370      PAGE ### JM
020380 FINAL: JM
020390      RETURN /* PRTLNE SUB. RETURN */ JM
020395 JM
020420 /****** JM
020430 /* JM
020440 /* THIS SUBROUTINE PRINTS THE READING GRADE LEVEL. JM
020450 /* JM
020460 /****** JM
020470 JM
020480 PRTRGL: JM
020490      IQ%=1 /* SUCCESS FLAG */ JM
020500      WC%=MAX(WC%,1) /* AT LEAST ONE WORD */ JM
020510      SC%=MAX(SC%,1) /* AT LEAST ONE SENTENCE */ JM
020520 JM
020530      /* KEEP OVERALL TOTALS */ JM
020540 JM
020560      XWC%=XWC%+WC% /* WORD COUNT TOTAL */ JM
020570      XSC%=XSC%+SC% /* SENTENCE COUNT TOTAL */ JM
020580      XSLC%=XSLC%+SLC% /* SYLLABLE COUNT TOTAL */ JM
020590      XUC%=XUC%+UW% /* UNCOMMON WORD TOTAL */ JM
020600      XLS%=XLS%+LS% /* LONG SENTENCE TOTAL */ JM
020610 JM
020620      SLC = SLC% : WC = WC% : SC = SC% JM
020630      NSW=SLC/WC /* NUMBER SYLLABLES PER WORD */ JM
020640      NWS=WC/SC /* NUMBER WORDS PER SENTENCE */ JM
020650 JM
020660      RGL=(NWS*.39)+(NSW*11.8)-15.59 /* COMPUTE READING LEVEL */ JM
020670      F1=UW% /* PERCENT OF UNCOMMON WORDS */ JM
020680      F2=LS% /* PERCENT OF LONG SENTENCES */ JM
020690      IF F1>0 THEN F1=F1/WC%*100 JM
020700      IF F2>0 THEN F2=F2/SC%*100 JM
020710 JM
020720      /* LET'S COMPUTE HEADER INFORMATION */ JM
020730 JM
020740      IF LAST$="Y" THEN BUFFER$="FINAL ANALYSIS" JM
020750      IF LAST$="Y" THEN JAPAN JM
020760      CONVERT PC%-1 TO ENDY$,PIC(###) JM
020770      BUFFER$="INTERIM ANALYSIS (PAGES " & BEGINY$ & " -" & ENDY$&")" JM
020790 JM
020800      /* NOW LET'S WRITE THIS INFORMATION OUT */ JM
020810 JM
020820 JAPAN: GOSUB TOP JM
020830      WRITE #12, USING H, C$ /* SKIP ONE LINE */ JM
020840      IF LAST$="Y" THEN WRITE #12,USING RQ,C$,BUFFER$ ! JM
020850      ELSE WRITE #12, USING RG, C$, BUFFER$ JM
020860      WRITE #12, USING H, C$ /* SKIP ONE LINE */ JM
020870      WRITE #12, USING RR, C$ /* PRINT "READABILITY RESULTS" */ JM

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 13:49 12/14/84 Page 37  
Input File is ANALYSIS in Library CRES@SRC on Volume CRES01

```

020880      WRITE #12, USING H, C$                /* SKIP ONE LINE */ JM
020890      WRITE #12, USING H2, C$                /* PRINT HEADING */ JM
020900      WRITE #12, USING H3, C$, SC$, WC$, SLCA /* PRINT DATA */ JM
020910      WRITE #12, USING H, C$                /* SKIP ONE LINE */ JM
020920      WRITE #12, USING H4, C$/* AVG. WORD/SENT. & SYL/WORD HEAD.*/ JM
020930      WRITE #12, USING H5, C$, NWS, NSW      /* PRINT DATA */ JM
020940      WRITE #12, USING H, C$                /* SKIP ONE LINE */ JM
020950      IF UNCOM$<>"Y" OR LNG_SENT$<>"Y" THEN JUMPY JM
020960      WRITE #12, USING H6, C$ /* % OF UW'S & % LONG SENT. HEAD. */ JM
020970      IF F2=100 THEN WRITE #12, USING H9, C$, F1, F2 ! JM
020980      ELSE WRITE #12, USING H7, C$, F1, F2 /* % OF UW'S & % LS */ JM
020990      GOTO JUMP2 JM
021000 JUMPY: IF UNCOM$<>"Y" THEN JUMP1 JM
021010      WRITE #12, USING A1, C$                /* % OF UW'S HEADING ONLY */ JM
021020      WRITE #12, USING A2, C$, F1            /* % OF UW'S */ JM
021030      GOTO JUMP2 JM
021040 JUMP1: IF LNG_SENT$<>"Y" THEN JUMP3 JM
021050      WRITE #12, USING B1, C$                /* % LONG SENT. HEAD. ONLY */ JM
021060      IF F2=100 THEN WRITE #12, USING A3, C$, F2 ! JM
021070      ELSE WRITE #12, USING A2, C$, F2      /* % LS */ JM
021080 JUMP2:WRITE #12, USING H, C$                /* SKIP ONE LINE */ JM
021090 JUMP3:WRITE #12, USING H8, C$,RGL          /* PRINT RGL */ JM
021100      GOTO JAKE JM
021110 A1:### Percent of Uncommon Words JM
021120 B1:### Percent of Long Sentences JM
021130 A2:### JM
021140 A3:### JM
021150 RR:### ----- READABILITY RESULTS -----! JM
021160 ----- JM
021170 RQ:### JM
021180 RG:### JM
021190 H:### JM
021200 H2:### Number of Sentences      Number of Words      Number of ! JM
021210 Syllables JM
021220 H3:### JM
021230 ## JM
021240 H4:### Avg. Number of Words per Sentence      Avg. Number of Sylla! JM
021250 bles per Word JM
021260 H5:### JM
021270 .## JM
021280 H6:### Percent of Uncommon Words      Percent of Long Sent! JM
021290 ences JM
021300 H7:### JM
021310 H9:### JM
021320 H8:### GRADE LEVEL (Based on DoD Readability Standard) = ##.## JM
021330 JAKE: /* CLEAR ALL VARIABLES FOR NEXT PART OF ANALYSIS */ JM
021340 JM
021350      CONVERT PC% TO BEGINY$,PIC(###) /* READY FOR NEXT TIME */ JM
021360      UW%,WC%,SC%,SLC%,LS% = 0 JM
021370      RETURN /* PRTRGL SUB. RETURN */ JM
021380 JM
021390 /***** JM
021400 /* JM
021410 /* SUBROUTINE TOP PRINTS A CENTERED TITLE AT THE TOP OF THE /* JM
021420 /* PAGE. /* JM
021430 /* JM
021440 /***** JM

```

[illegible]

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 13:49 12/14/84 Page 39  
Input File is ANALYSIS in Library CRES0SRC on Volume CRES01

```

022000 PREQ JM
022010 FINICH: MAT UCWCT0= ZER /* ZERO OUT ALL COUNTS */ JM
022020 INIT(" ")UCW$() /* BLANK OUT OLD UNCOMMON LIST */ JM
022030 JM
022040 NINETY: JM
022050 RETURN /* UWL SUBROUTINE RETURN */ JM
022060 JM
022070 /****** JM
022080 /* JM
022090 /* THIS SUBROUTINE STRIPS PUNCTUATION FOR THE WORD WORD1$. JM
022100 /* JM
022110 /****** JM
022120 JM
022130 SPUNK: JM
022140 BS$=SENTPOS$ : BC$=CLSPOS$ : BB$=" " JM
022150 FOUN0=0 /* WE HAVEN'T FOUND ABBRE. YET */ JM
022160 I0 = 1 /* ASSUME SUCCESS FOR NOW */ JM
022170 INVLDS$=" " /* CLEAR INVALID WORD FLAG */ JM
022180 JM
022200 IF SENTPOS$ = "E" THEN SENTPOS$ = "B" /* START OF SENT. */ JM
022230 IF CLSPQS$ = "E" THEN CLSPOS$ = "B" /* START OF NEW CLAUSE */ JM
022240 JM
022250 LEAD_PUNC: /* CHECK FOR LEADING PUNCTUATION */ JM
022260 IF LEN0=1 THEN SGLCHAR /* END OF WORD */ JM
022270 AS=STR(WORD1$,1,1) JM
022280 IF (AS < "0" ) OR (AS > "9" AND AS < "A") OR ! JM
022290 (AS > "Z" AND AS < "a") ! JM
022300 THEN INVLCHR JM
022310 JM
022320 IF AS <= "z" THEN CHKTRLG JM
022330 JM
022340 INVLCHR: /* INVALID LEADING CHARACTER */ JM
022350 IF AS = "(" THEN NEG0(3) = 1 /* NEGATIVE SWITCH */ JM
022360 WORD1$ = STR(WORD1$,2) JM
022370 LEN0 = LEN0 - 1 JM
022380 GOTO LEAD_PUNC JM
022390 JM
022400 CHKTRLG: /* CHECK FOR TRAILING PUNCTUATION */ JM
022410 IF LEN0=1 THEN SGLCHAR /* END OF WORD */ JM
022420 AS=STR(WORD1$,LEN0,1) JM
022430 IF (AS < "0" ) OR (AS > "9" AND AS < "A") OR ! JM
022440 (AS > "Z" AND AS < "a") ! JM
022450 THEN INVTCHR JM
022460 JM
022470 IF AS <= "z" THEN CHKNWSENT JM
022480 JM
022490 INVTCHR: /* INVALID TRAILING CHARACTER */ JM
022500 IF AS = ")" THEN SETNEG JM
022510 J0 = POS(".,!?" = AS) JM
022520 IF J0=0 THEN BAR JM
022530 IF AS<> "." THEN ENDSNT /* CAN'T BE ABBRE. */ JM
022540 WORDQ$=STR(WORD1$,1,LEN0-1) /* GET RID OF THE PERIOD */ JM
022550 TRAN( WORDQ$, ! JM
022560 "AaBbCcDdEeFfGgHhIiJjKkLlMmNnOoPpQqRrSsTtUuVvWwXxYyZz" ) ! JM
022570 REPLACING JM
022580 IF ABBRE0=0 THEN ENDSNT JM
022590 FOR QD0=1 TO ABBRE0 JM

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 13:49 12/14/84 Page 40  
Input File is ANALYSIS in Library CRES@SRC on Volume CRES01

```

022600 IF STR(WORDQ$,1,LEN%) > STR(ABBRE$(QD%),1,LEN%) THEN QD%=ABBRE% JM
022610 IF STR(WORDQ$,1,LEN%) <> STR(ABBRE$(QD%),1,LEN%) THEN DROP JM
022620 IF LEN(ABBRE$(QD%)) > LEN%+1 THEN DROP JM
022630 POUN%=1 JM
022640 QD%=ABBRE% JM
022650 DROP: NEXT QD% JM
022660 IF POUN%=0 THEN ENDSNT JM
022670 BAR: JM
022680 IF A$ = ":" OR A$ = "," THEN SETCLS JM
022690 GOTO RMV_CHR JM
022700 JM
022710 SETNEG: /* SET NEGATIVE SWITCH */ JM
022720 NEG%(3) = 0 JM
022730 GOTO RMV_CHR JM
022740 JM
022750 ENDSNT: JM
022760 SENTPOS$ = "E" /* SET END OF SENTENCE FLAG */ JM
022770 IF J% = 1 THEN BB$ = "Y" JM
022780 JM
022790 SETCLS: JM
022800 IF A$="," THEN LIST%=LIST%+1 /* COUNT COMMAS FOR 'LIST?' */ JM
022810 CLSPOS$ = "E" /* SET END OF CLAUSE FLAG */ JM
022820 JM
022830 RMV_CHR: /* REMOVE THE LAST CHAR. OF WORD */ JM
022840 STR(WORD1$,LEN%,1) = " " JM
022850 LEN% = LEN% - 1 JM
022860 GOTO CHKTRLG JM
022870 JM
022880 SGLCHAR: /* CHECK FOR VALID SINGLE CHARACTER, 'A' OR 'I' OR 'a' */ JM
022890 IF BB$ = "Y" THEN SAV1 JM
022900 A$ = STR(WORD1$,1,1) JM
022910 IF A$ = "I" OR A$ = "A" THEN SAV2 /* BEG. OF SENTENCE */ JM
022920 IF A$ = "a" THEN SP_RET JM
022930 GOTO SAV3 JM
022940 JM
022950 SAV1: /* SAVE SENT. AND CLS. FLAGS */ JM
022960 SENTPOS$ = BS$ JM
022970 CLSPOS$ = BC$ JM
022980 GOTO SAV3 JM
022990 JM
023000 SAV2: /* CHECK AND SAVE SENT. FLAG */ JM
023010 IF SENTPOS$ = "B" THEN SENTPOS$ = " " JM
023020 GOTO SP_RET JM
023030 JM
023040 CHKNWSNT: /* CHECK FOR START OF NEW SENTENCE */ JM
023050 A$ = STR(WORD1$,LEN%,1) JM
023060 /* CHECK FOR " 'S " OR " 's " */ JM
023070 IF A$="s" THEN A$="S" JM
023080 IF A$="S" AND STR(WORD1$,LEN%-1,1) = "' " ! JM
023090 THEN LEN% = LEN% - 2 JM
023100 A$ = STR(WORD1$,1,1) JM
023120 IF A$ < "A" THEN SAV3 JM
023130 IF A$ > "Z" THEN CHK_NUM JM
023140 GOTO CHK_NUM JM
023150 JM
023160 /* CHECK FOR ACRONYMS, PROPER NOUNS */ JM
023170 IF A$ >= "a" THEN CHK_NUM JM

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 13:49 12/14/84 Page 41  
Input File is ANALYSIS in Library CRES@SRC on Volume CRES01

```

023180 IF BB$ = "Y" THEN BADWD JM
023190 GOTO SAV3 JM
023200 JM
023210 CHK_NUM: /* CHECK FOR IMBEDDED NUMBER OR NUMBERS */ JM
023220 J% = POS( STR(WORD1$,1,LEN%) < "A" ) JM
023230 IF J% = 0 THEN SP_RET JM
023240 IF STR(WORD1$,J%,1)<>"." THEN SAV3 JM
023250 JM
023260 BADWD: /* SET BAD WORD FLAG */ JM
023270 INVLD$ = "P" JM
023280 I% = 0 JM
023290 GOTO SP_RET JM
023300 JM
023310 SAV3: /* SAVE AND SET FLAGS */ JM
023320 INVLD$ = "Y" JM
023330 IF AS < "A" AND LEN% < 3 AND SENTPOSS = "E" THEN I% = 0 JM
023340 JM
023350 SP_RET: RETURN /* STRIP PUNC. SUB. RETURN */ JM
023360 JM
023370 /****** JM
023380 /* JM
023390 /* THIS SUBROUTINE PRINTS ALL WORDS AND FLAGS. JM
023400 /* JM
023410 /****** JM
023420 JM
023430 PRNT_WDS: JM
023440 JM
023450 IF PHCT% = 1 THEN NORML JM
023460 SAVPH$(PHCT%)=WORDS$ /* SAVE THE LAST WORD */ JM
023470 SAVPHL$(PHCT%)=LW$(1) /* SAVE THE LAST WORD'S LENGTH */ JM
023480 SAVSYL$(PHCT%)=TLC% /* SAVE THE SYLLABLE COUNT */ JM
023490 IF WSL$="Y" THEN SAVRECL$(PHCT%)=RECL$ JM
023500 IF WSL$="Y" THEN WSL$="N" JM
023510 WORDS$=SAVPH$(1) /* USE THE FIRST WORD */ JM
023520 LW$(1)=SAVPHL$(1) /* USE THE 1ST WORD'S LENGTH */ JM
023530 TLC%=SAVSYL$(1) /* USE THE 1ST WORD'S SYL. CT. */ JM
023540 IF SAVSUBL$(1)<>0 THEN STR(WORDS$,25,35)=STR(SAVRECL$(1),21,35) JM
023550 FOR Q% = 1 TO PHCT%-1 /* MOVE WORDS IN THE ARRAY */ JM
023560 SAVPH$(Q%)=SAVPH$(Q%+1) JM
023570 SAVPHL$(Q%)=SAVPHL$(Q%+1) JM
023580 SAVSYL$(Q%)=SAVSYL$(Q%+1) JM
023590 SAVSUB$(Q%)=SAVSUB$(Q%+1) JM
023600 SAVSUBL$(Q%)=SAVSUBL$(Q%+1) JM
023610 SAVRECL$(Q%)=SAVRECL$(Q%+1) JM
023620 NEXT Q% JM
023630 SAVSUBL$(PHCT%)=0 JM
023640 K4%=PHCT% :GOTO LAXOPI JM
023650 NORML: K4%=IW% JM
023660 IF CW$="X" THEN K4%=1 :IF CW$="Y" THEN LNPAR JM
023670 IF K4%=0 THEN LNPAR /* K4% = # OF WORDS */ JM
023680 LAXOPI: FOR I%=1 TO K4% JM
023690 IF I% = 1 THEN SKIP1 JM
023700 PHCT% = PHCT%-1 JM
023710 WORDS$=SAVPH$(I) JM
023720 LW$(I%)=SAVPHL$(I) JM
023730 TIA%=SAVSYL$(I) JM
023740 IF SAVSUBL$(I)<>0 THEN STR(WORDS$,25,35)= : JM

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 13:49 12/14/84 Page 42  
Input File is ANALYSIS in Library CRES0SRC on Volume CRES01

```

023750      STR(SAVRECL$(1),21,35) JM
023760      IF PHCT%=1 THEN SKIPT JM
023770      FOR Q% = 1 TO PHCT%-1 /* MOVE WORDS IN THE ARRAY */ JM
023780          SAVSYL$(Q%)=SAVSYL$(Q%+1) JM
023790          SAVPH$(Q%)=SAVPH$(Q%+1) JM
023800          SAVPHL$(Q%)=SAVPHL$(Q%+1) JM
023810          SAVSUB$(Q%)=SAVSUB$(Q%+1) JM
023820          SAVSUBL$(Q%)=SAVSUBL$(Q%+1) JM
023830          SAVRECL$(Q%)=SAVRECL$(Q%+1) JM
023840      NEXT Q% JM
023850      SAVSUBL$(PHCT%)=0 JM
023860 SKIPT: IF D%=2 THEN SUBT /* FLAGS ONLY */ JM
023870 JM
023880 /* COMPUTE THE LENGTH OF DATA TO PRINT (IF IT'S TOO LONG FOR */ JM
023890 /* THIS LINE, PRINT THE LINE AND PUT DATA ON NEXT LINE) */ JM
023900 JM
023910      J% = LW$(I%) JM
023920      K1% = 0 JM
023930      K2% = 0 JM
023940      K3% = 1 JM
023950 JM
023960      /* CHECK FOR UNCOMMON WORD */ JM
023970 JM
023980      IF UCWFLG$(I%) = "Y" AND UNCOM$="Y" ! JM
023990          THEN PRTUNC /* UNC WORD AND OPTION SELECTED */ JM
024000 JM
024010      /* IT'S A REGULAR WORD - CHECK FOR SUBSTITUTES */ JM
024020 JM
024030      IF NUM_SUB$="0" OR STR(WORDS$,25,1)=" " OR INVLD$="Y" : JM
024035          THEN CHKLL /* NO SUBS.*/ JM
024040      K1% = -1 JM
024050      K1% = POS(STR(WORDS$,25,20) = " ") /* ?????? WORDS ?????*/ JM
024060      J% = J% + K1% + 6 /* ADD SPACES FOR '<...>(<...*)' */ JM
024070 JM
024080      /* CHECK ON SECOND SUBSTITUTE */ JM
024090 JM
024100      IF NUM_SUB$="1" OR STR(WORDS$,45,1)=" " THEN CHKLL JM
024110      /* ONLY ONE SUBSTITUTE NEEDED OR NO 2ND SUBS. */ JM
024120 JM
024130      K2% = -1 JM
024140      K2% = POS(STR(WORDS$,45,15)=" ") JM
024150      J% = J% + K2% + 1 /* ADD SPACE FOR '/' */ JM
024160 JM
024170      /* CHECK ON LINE LENGTH */ JM
024180 JM
024190 CHKLL: IF IO% + J% > LL% THEN GOSUB PRTLNE /* PRINT LINE */ JM
024200      IF K3% = 0 THEN OUT /* ERROR FROM PRTLNE */ JM
024210 JM
024220      /* BUILD OUTPUT LINE */ JM
024230 JM
024240      IF K1% = 0 THEN PIT /* NO SUBSTITUTES FOR WORD */ JM
024250      STR(OLINE$,IO%,1) = "<" JM
024260      IO% = IO% + 1 JM
024270 JM
024280      /* NOW ADD THE SUBSTITUTES */ JM
024290 JM
024300 PIT: STR(OLINE$,IO%,LW$(I%)) = STR(WORDS$,1,LW$(I%)) JM

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 13:49 12/14/84 Page 43  
Input File is ANALYSIS in Library CRES@SRC on Volume CRES01

```

024310      WC1%=WC1%+1                      /* INCREMENT THE WORD COUNT */ JM
024320      WP1%=WP1%+1                      /* INCREMENT THE WORDS PER PAR. COUNT */ JM
024330      SLC1%=SLC1%+TLC%                 /* INCREMENT THE SYLLABLE COUNT */ JM
024360      IO% = IO% + LW%(I%)              JM
024370      IF K1% = 0 THEN SUBT              /* STILL NO SUBSTITUTES */ JM
024380                                           JM
024390      /* INSERT FIRST SUBSTITUTE */      JM
024400                                           JM
024410      STR(OLINE$,IO%,3) = "><*"        JM
024420      IO% = IO% + 3                    JM
024430      STR(OLINE$,IO%,K1%) = STR(WORD$$,25,K1%) JM
024440      IO% = IO% + K1%                  JM
024450                                           JM
024460      /* CHECK FOR SECOND SUBSTITUTE */  JM
024470                                           JM
024480      IF K2% = 0 THEN FN$H              JM
024490      STR(OLINE$,IO%,1) = "/"          JM
024500      IO% = IO% + 1                    JM
024510      STR(OLINE$,IO%,K2%) = STR(WORD$$,45,K2%) JM
024520      IO% = IO% + K2%                  JM
024530                                           JM
024540      /* FINISH THE ENTRY */            JM
024550                                           JM
024560 FN$H:  STR(OLINE$,IO%,2) = "*"       JM
024570      IO% = IO% + 2                    JM
024580      GOTO SUBT                         JM
024590                                           JM
024600      /* PRINT UNCOMMON WORDS */        JM
024610                                           JM
024620 PRTUNC: J% = J% + 2                  /* ADD [...] */ JM
024630                                           JM
024640      /* CHECK LINE LENGTH */          JM
024650                                           JM
024660      PFLAG% = K3%                     JM
024670      IF IO% + J% > LL% THEN GOSUB PRTLNE JM
024680      IF PFLAG% = 0 THEN OUT             /* ERROR RETURN */ JM
024690      STR(OLINE$,IO%,1) = "["           JM
024700      IO% = IO% + 1                    JM
024710      L$=STR(WORD$$,LW%(I%),1)          JM
024720      PZ%=POS("?:.,!"=L$)              JM
024730      IF PZ%>0 THEN LW%(I%)=LW%(I%)-1  JM
024740      STR(OLINE$,IO%,LW%(I%)) = STR(WORD$$,1,LW%(I%)) JM
024750      WC1%=WC1%+1                      /* INCREMENT THE WORD COUNT */ JM
024760      WP1%=WP1%+1                      /* INCREMENT THE WORDS PER PAR. COUNT */ JM
024770      SLC1%=SLC1%+TLC%                 /* INCREMENT THE SYLLABLE COUNT */ JM
024800      IO% = IO% + LW%(I%)              JM
024810      STR(OLINE$,IO%,1) = "]" : IO%=IO%+1 : UCWFLG$(I%) = " " JM
024820      IF PZ%>0 THEN STR(OLINE$,IO%,1) = STR(WORD$$,LW%(I%)+1,1) JM
024830      IF PZ%>0 THEN IO%=IO%+1          JM
024840 SUBT:  IF EOS%=0 THEN L$ENT /*HAVEN'T FOUND END OF SENTENCE */ JM
024850      WC% = WC% + WC1%                  /* INCREMENT WORD COUNT */ JM
024860      WP% = WP% + WP1%                  /* INCREMENT WORDS/PAR COUNT */ JM
024870      SLC% = SLC% + SLC1%               /* INCREMENT SYLLABLE COUNT */ JM
024900      EOS%, WC1%, WP1%, SLC1% = 0      /* RESET BUFFER VARIABLES */ JM
024910      IF UNCP%>1 THEN GOSUB ADD_UNCWL /* ADD UNCOMMON WORDS */ JM
024920                                           JM
024930      /* PRINT THE FLAGS - CHECK LONG SENTENCE FLAG */ JM

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 13:49 12/14/84 Page 44  
Input File is ANALYSIS in Library CRES@SRC on Volume CRES01

```

024940 JM
024950 LSENT:IF LSENT(I%)=0 OR LNG_SEN<>"Y" THEN PASSF /*NO LONG SEN.*/ JM
024960 PFLAG% = K3% JM
024970 IF IO% + 7 > LL% THEN GOSUB PRTLNE JM
024980 IF PFLAG% = 0 THEN OUT /* ERROR RETURN */ JM
024990 CONVERT LSENT(I%) TO K1$, PIC (###) JM
025000 STR(OLINE$,IO%,7) = "<*% K1$ & %*" JM
025010 IO% = IO% + 7 :LSENT(I%)=0 JM
025020 IF (LIST%(3 AND CONSI% = 0) OR LIST%(5 THEN PASSF JM
025030 PFLAG% = K3% JM
025040 IF IO% + 11 > LL% THEN GOSUB PRTLNE JM
025050 IF PFLAG% = 0 THEN OUT /* ERROR RETURN */ JM
025060 STR(OLINE$,IO%,11) = "<* LIST? %*" JM
025070 IO% = IO% + 11 JM
025080 JM
025090 /* CHECK ON PASSIVE FLAG */ JM
025100 JM
025110 PASSF: IF PASSV<>"Y" AND TRNK_PV<>"Y" THEN DNEG JM
025120 IF PASSV(I%)<>"Y" THEN DNEG JM
025130 PFLAG% = K3% JM
025140 IF IO% + 19 > LL% THEN GOSUB PRTLNE JM
025150 IF PFLAG% = 0 THEN OUT /* ERROR RETURN */ JM
025160 STR(OLINE$,IO%,19) = "<* PASSIVE VOICE %*" JM
025170 IO% = IO% + 19 :PASSV(I%) = " " JM
025180 JM
025190 /* CHECK DOUBLE NEGATIVE FLAG */ JM
025200 JM
025210 DNEG: IF NEG<>"Y" THEN PREP /* NO DOUBLE NEGATIVE OPTION */ JM
025220 IF DNFL(I%)<>"Y" THEN PREP JM
025230 PFLAG% = K3% JM
025240 IF IO% + 21 > LL% THEN GOSUB PRTLNE JM
025250 IF PFLAG% = 0 THEN OUT /* ERROR RETURN */ JM
025260 STR(OLINE$,IO%,21) = "<* DOUBLE NEGATIVE %*" JM
025270 IO% = IO% + 21 :DNFL(I%) = " " JM
025370 JM
025480 /* CHECK PREPOSITION FLAG */ JM
025490 JM
025500 PREP: IF PREP_CNT<>"Y" THEN FUTURE JM
025510 IF PREPFL(I%) = 0 THEN FUTURE JM
025520 PFLAG% = K3% JM
025530 IF IO% + 33 > LL% THEN GOSUB PRTLNE JM
025540 IF PFLAG% = 0 THEN OUT /* ERROR RETURN */ JM
025550 CONVERT PREPFL(I%) TO OOS$, PIC (###) JM
025560 STR(OLINE$,IO%,33) = "<* TOO MANY (" & OOS$ & ") PREPOSI: JM
025570 TIONS %*" JM
025580 PREPFL(I%) = 0 /* RESET PREP. FLAG */ JM
025590 IO% = IO% + 33 JM
025600 JM
025610 /* LEAVE ROOM FOR FUTURE EXPANSION */ JM
025620 JM
025630 FUTURE: IO% = IO% + 1 JM
025640 NEXT I% /* THIS INCREMENTS THE BIG LOOP OF THIS ROUTINE */ JM
025650 JM
025660 /* NOW CHECK UNIQUE FLAGS - CHECK LONG PARAGRAPH FLAG */ JM
025670 JM
025680 LNPAR: IF END_O_PAR$ <> "Y" THEN PERMSG JM
025690 END_O_PAR$ = " " /* RESET END_O_PAR FLAG */ JM

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 13:49 12/14/84 Page 45  
Input File is ANALYSIS in Library CRES@SRC on Volume CRES01

```

025700      IF LNG_PAR$<>"Y" THEN SPRST /* PRINT OUT THE PARAGRAPH */ JM
025710      IF SP%<LNGSNT% AND WP%<LNGWDS% : JM
025720      THEN SPRST /* NOT TOO LONG */ JM
025730      PFLAG% = K3% JM
025740      IF IO% + 57 > LL% THEN GOSUB PRTLNE JM
025750      IF PFLAG% = 0 THEN OUT JM
025760      CONVERT SP% TO OO$, PIC (###) JM
025770      CONVERT WP% TO OI$, PIC (###) JM
025780      STR(OLINE$,IO%,57) = "<(* PARAGRAPH TOO LONG (" & OO$ & : JM
025790      " SENTENCES WITH " & OI$ & " WORDS ) *>" JM
025800      IO% = IO% + 57 : SP%,WP% = 0 JM
025810      GOTO PERMSG JM
025820 SPRST: WP%,SP% = 0 JM
025830      PFLAG% = K3% JM
025840      IF IO% > 1 THEN GOSUB PRTLNE JM
025850      IF PFLAG% = 0 THEN OUT JM
025860      JM
025870      /* CHECK FOR PERSONAL SENTENCE MESSAGE; */ JM
025880 /* NO NEED FOR THIS CODE IN PRT_WDS BECAUSE THE LAST WORD */ JM
025890 /* CAN ONLY HAPPEN HERE. */ JM
025900 JM
025910 PERMSG: IF E_O_P$<>"Y" OR PERS_SENT$<>"Y" THEN RESET JM
025920      K1 = 0 JM
025930      IF INT_ANL$<>"Y" THEN SC=SC% JM
025940      IF INT_ANL$="Y" THEN SC=SC%+XSC% JM
025950      IF SC=0 THEN K1=0 JM
025960      IF SC=0 THEN NOSNT JM
025970      PE=PE% JM
025980      K1=PE*100.0/SC JM
025990      NOSNT: K3% = 1 JM
026000      PFLAG% = K3% JM
026010      IF IO% + 30 > LL% THEN GOSUB PRTLNE JM
026020      IF PFLAG% = 0 THEN OUT JM
026030      STR(OLINE$,IO%,2) = "<*" JM
026040      IO%=IO%+3 JM
026050      IF K1<100 THEN CONVERT K1 TO STR(OLINE$,IO%,4), PIC(###.##) JM
026060      ELSE CONVERT K1 TO STR(OLINE$,IO%,5), PIC(###.###) JM
026070      IF K1<100 THEN IO%=IO%+4 ELSE IO%=IO%+5 JM
026080      STR(OLINE$,IO%,23) = "% PERSONAL SENTENCES *)" JM
026090      IO%=IO%+23 JM
026100      GOSUB PRTLNE JM
026110      IF K3% = 0 THEN OUT JM
026120      JM
026130      /* RESET WORK AREAS */ JM
026140      JM
026150 RESET: IF CW$ = "X" THEN REMOVE JM
026160      WORDS$( ) = ALL(HEX(20)) JM
026170      IW% = 0 JM
026180      GOTO POSTPONE JM
026190      JM
026200      /* REMOVE FIRST WORD */ JM
026210      JM
026220 REMOVE: IF IW% <= 1 THEN SINGLE JM
026230      FOR I% = 1 TO (IW%-1) JM
026240          WORDS$(I%) = WORDS$(I%+1) JM
026250          PASSVF$(I%) = PASSVF$(I%+1) JM
026260          DNFL$(I%) = DNFL$(I%+1) JM

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 13:49 12/14/84 Page 46  
Input File is ANALYSIS in Library CRES@SRC on Volume CRES01

```

026270      NOUNFL$(I%) = NOUNFL$(I%+1)      JM
026280      UCWFLG$(I%) = UCWFLG$(I%+1)      JM
026290      LCLSFL$(I%) = LCLSFL$(I%+1)      JM
026300      LSENT$(I%) = LSENT$(I%+1)      JM
026310      PREPFL$(I%) = PREPFL$(I%+1)      JM
026320      LW$(I%) = LW$(I%+1)      JM
026330      NEXT I%      JM
026340      WORDS$(IW%) = ALL(HEX(20))      JM
026350      PASSVP$(IW%) = " "      JM
026360      DNFL$(IW%) = " "      JM
026370      NOUNFL$(IW%) = " "      JM
026380      UCWFLG$(IW%) = " "      JM
026390      LCLSFL$(IW%) = 0      JM
026400      LSENT$(IW%) = 0      JM
026410      PREPFL$(IW%) = 0      JM
026420      LW$(IW%) = 0      JM
026430      GOTO POSTPONE      JM
026440 SINGLE: WORDS$(1) = ALL(HEX(20))      JM
026450      PASSVP$(1) = " "      JM
026460      DNFL$(1) = " "      JM
026470      NOUNFL$(1) = " "      JM
026480      UCWFLG$(1) = " "      JM
026490      LCLSFL$(1) = 0      JM
026500      LSENT$(1) = 0      JM
026510      PREPFL$(1) = 0      JM
026520      LW$(1) = 0      JM
026530 POSTPONE:      JM
026540      IF IW% > 0 THEN IW% = IW%-1      JM
026550      IF D% = 3 THEN OUT      JM
026560      JM
026570      /* CHECK ON UNCOMMON WORDS AND PRINT IF TABLE IS FULL */      JM
026580      JM
026590      IF IDX% < 200 THEN OUT      JM
026600      GOSUB UWL      JM
026601      GOSUB TOP      JM
026610 OUT: IF CW$="Y" THEN CW$=" "      JM
026620      IF WC1%=0 THEN LIST%, CONSIS%=0      JM
026630      RETURN      /* PRNT_WDS SUB. RETURN */      JM
026640      JM
026650 /******      JM
026660 /*      JM
026670 /* THIS SUBROUTINE PRINTS ALL WORDS AND FLAGS FROM CHKPHSFL.      JM
026680 /* THIS SUB. IS SIMILAR TO PRNT_WDS, BUT ONLY CALLED WHEN      JM
026690 /* WORDS ARE REMOVED FROM POSSIBLE PHRASES.      JM
026700 /*      JM
026710 /******      JM
026720      JM
026730 PRT_WDS:      JM
026740      IF PHCT% <> 1 THEN SKIPTI      JM
026750      WORDZ$=WORDO$      JM
026760      LENO%=LW%(1)      JM
026770      TLO%=TLC%      JM
026780      GOTO SKIPTO      JM
026790 SKIPTI: PHCT% = PHCT%-1      JM
026800      WORDZ$=SAVPH$(1)      JM
026810      LENO%=SAVPHL%(1)      JM
026820      TLO%=SAVSYL%(1)      JM

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 13:49 12/14/84 Page 47  
Input File is ANALYSIS in Library CRESESRC on Volume CRES01

```

026830      IF SAVSUBL(1)<>0 THEN STR(WORDZ$,25,35)=          : JM
026840      STR(SAVRECL$(1),21,35)                               JM
026850      IF PHCT=1 THEN SKIPTO                                  JM
026860      FOR Q% = 1 TO PHCT-1      /* MOVE WORDS IN THE ARRAY */ JM
026870          SAVSYL(Q%)=SAVSYL(Q%+1)                           JM
026880          SAVPH$(Q%)=SAVPH$(Q%+1)                             JM
026890          SAVPHL(Q%)=SAVPHL(Q%+1)                             JM
026900          SAVSUB$(Q%)=SAVSUB$(Q%+1)                           JM
026910          SAVSUBL(Q%)=SAVSUBL(Q%+1)                           JM
026920          SAVRECL$(Q%)=SAVRECL$(Q%+1)                         JM
026930      NEXT Q%                                                JM
026940 SKIPTO: SAVSUBL(PHCT)=0                                    JM
026950      IF D=2 THEN LSENT0      /* FLAGS ONLY */             JM
026960                                                                JM
026970 /* COMPUTE THE LENGTH OF DATA TO PRINT (IF IT'S TOO LONG FOR */ JM
026980 /* THIS LINE, PRINT THE LINE AND PUT DATA ON NEXT LINE) */ JM
026990                                                                JM
027000      J% = LENO%                                              JM
027010      K1% = 0                                                JM
027020      K2% = 0                                                JM
027030      K3% = 1                                                JM
027040                                                                JM
027050      /* CHECK FOR UNCOMMON WORD */                          JM
027060                                                                JM
027070      IF UCWPLG$(I%) = "Y" AND UNCOM$="Y"                    : JM
027080          THEN PRTUNCO      /* UNC WORD AND OPTION SELECTED */ JM
027090      /* IT'S A REGULAR WORD - CHECK FOR SUBSTITUTES */      JM
027100                                                                JM
027110      IF NUM_SUB$="0" OR STR(WORDZ$,25,1)=" " THEN CHKLL0/* NO SUBS.*/ JM
027120          K1% = -1                                             JM
027130          K1% = POS(STR(WORDZ$,25,20) = " ") /* ?????? WORDS ?????*/ JM
027140          J% = J% + K1% + 6      /* ADD SPACES FOR '<...>(*...*)' */ JM
027150                                                                JM
027160          /* CHECK ON SECOND SUBSTITUTE */                      JM
027170                                                                JM
027180          IF NUM_SUB$="1" OR STR(WORDZ$,45,1)=" " THEN CHKLL0 JM
027190          /* ONLY ONE SUBSTITUTE NEEDED OR NO 2ND SUBS. */    JM
027200                                                                JM
027210          K2% = -1                                             JM
027220          K2% = POS(STR(WORDZ$,45,15) = " ")                  JM
027230          J% = J% + K2% + 1      /* ADD SPACE FOR '/' */      JM
027240                                                                JM
027250          /* CHECK ON LINE LENGTH */                          JM
027260                                                                JM
027270 CHKLL0: IF IO% + J% > LL% THEN GOSUB PRTLNE      /* PRINT LINE */ JM
027280      IF K3% = 0 THEN OUT0      /* ERROR FROM PRTLNE */      JM
027290                                                                JM
027300          /* BUILD OUTPUT LINE */                             JM
027310                                                                JM
027320          IF K1% = 0 THEN PITO      /* NO SUBSTITUTES FOR WORD */ JM
027330          STR(OLINE$,IO%,1) = "<"                               JM
027340          IO% = IO% + 1                                         JM
027350                                                                JM
027360          /* NOW ADD THE SUBSTITUTES */                       JM
027370                                                                JM
027380 PITO: STR(OLINE$,IO%,LENO%) = STR(WORDZ$,1,LENO%)          JM
027390      WCL% = WCL% + 1      /* INCREMENT THE WORD COUNT */    JM

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 13:49 12/14/84 Page 48  
Input File is ANALYSIS in Library CRES\$SRC on Volume CRES01

```

027400      WPl% = WPl% + 1          /* INCREMENT THE WORDS PER PAR. COUNT */ JM
027410      SLC1% = SLC1% + TLO%    /* INCREMENT THE SYLLABLE COUNT */ JM
027420      IO% = IO% + LENO%      JM
027430      IF K1% = 0 THEN LSENTO  /* STILL NO SUBSTITUTES */ JM
027440      JM
027450      /* INSERT FIRST SUBSTITUTE */ JM
027460      JM
027470      STR(OLINE$, IO%, 3) = "<*" JM
027480      IO% = IO% + 3           JM
027490      STR(OLINE$, IO%, K1%) = STR(WORDZ$, 25, K1%) JM
027500      IO% = IO% + K1%         JM
027510      JM
027520      /* CHECK FOR SECOND SUBSTITUTE */ JM
027530      JM
027540      IF K2% = 0 THEN FNSHO    JM
027550      STR(OLINE$, IO%, 1) = "/" JM
027560      IO% = IO% + 1           JM
027570      STR(OLINE$, IO%, K2%) = STR(WORDZ$, 45, K2%) JM
027580      IO% = IO% + K2%        JM
027590      JM
027600      /* FINISH THE ENTRY */ JM
027610      JM
027620      FNSHO: STR(OLINE$, IO%, 2) = "*" JM
027630      IO% = IO% + 2          JM
027640      GOTO LSENTO            JM
027650      JM
027660      /* PRINT UNCOMMON WORDS */ JM
027670      JM
027680      PRTUNCO: J% = J% + 2   /* ADD [...] */ JM
027690      JM
027700      /* CHECK LINE LENGTH */ JM
027710      JM
027720      PFLAG% = K3%           JM
027730      IF IO% + J% > LL% THEN GOSUB PRTLNE JM
027740      IF PFLAG% = 0 THEN OUTO /* ERROR RETURN */ JM
027750      STR(OLINE$, IO%, 1) = "[" JM
027760      IO% = IO% + 1           JM
027770      L$ = STR(WORDZ$, LENO%, 1) JM
027780      PZ% = POS(" ; : ? . , ! " = L$) JM
027790      IF PZ% > 0 THEN LENO% = LENO% - 1 JM
027800      STR(OLINE$, IO%, LENO%) = STR(WORDZ$, 1, LENO%) JM
027810      WCl% = WCl% + 1        /* INCREMENT THE WORD COUNT */ JM
027820      WPl% = WPl% + 1        /* INCREMENT THE WORDS PER PAR. COUNT */ JM
027830      SLC1% = SLC1% + TLO%    /* INCREMENT THE SYLLABLE COUNT */ JM
027840      IO% = IO% + LENO%      JM
027850      STR(OLINE$, IO%, 1) = "]" : IO% = IO% + 1 : UCWFLG$(1%) = " " JM
027860      IF PZ% > 0 THEN STR(OLINE$, IO%, 1) = STR(WORDZ$, LENO% + 1, 1) JM
027870      IF PZ% > 0 THEN IO% = IO% + 1 JM
027880      JM
027890      /* PRINT THE FLAGS - CHECK LONG SENTENCE FLAG */ JM
027900      JM
027910      LSENTO: IF LSENT%(1%) = 0 THEN PASSFO /* NO LONG SENT. */ JM
027920      IF LNG_SEN$("<") = "Y" THEN PASSFO /* LONG SEN. OPTION NOT Y */ JM
027930      PFLAG% = K3%           JM
027940      IF IO% + 7 > LL% THEN GOSUB PRTLNE JM
027950      IF PFLAG% = 0 THEN OUTO /* ERROR RETURN */ JM
027960      CONVERT LSENT%(1%) TO K1$, PIC (###) JM

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 13:49 12/14/84 Page 49  
Input File is ANALYSIS in Library CRES@SRC on Volume CRES01

```

027970      STR(OLINE$,IO%,7)="<<*& K1$ & *>"          JM
027980      IO% = IO% + 7 :LSENT%(I%)=0                  JM
027990      IF (LIST%(3 AND CONSI% = 0) OR LIST%(5 THEN PASSFO      JM
028000      PFLAG% = K3%                                    JM
028010      IF IO% + 11 > LL% THEN GOSUB PRTLNE             JM
028020      IF PFLAG% = 0 THEN OUT                          /* ERROR RETURN */ JM
028030      STR(OLINE$,IO%,11)="<<*& LIST? *>"          JM
028040      IO% = IO% + 11                                    JM
028050                                                    JM
028060      /* CHECK ON PASSIVE FLAG */                     JM
028070                                                    JM
028080 PASSFO: IF PASSV$(<)"Y" AND TRNK_PV$(<)"Y" THEN DNEGO      JM
028090      IF PASSVF$(I%) <> "Y" THEN DNEGO                  JM
028100      PFLAG% = K3%                                    JM
028110      IF IO% + 19 > LL% THEN GOSUB PRTLNE             JM
028120      IF PFLAG% = 0 THEN OUTO                          /* ERROR RETURN */ JM
028130      STR(OLINE$,IO%,19) = "<*& PASSIVE VOICE >"      JM
028140      IO% = IO% + 19 :PASSVF$(IW%)=" "                JM
028150                                                    JM
028160      /* CHECK DOUBLE NEGATIVE FLAG */                 JM
028170                                                    JM
028180 DNEGO: IF NEGS$(<)"Y" THEN PREPO                    JM
028190      IF DNFL$(I%) <> "Y" THEN PREPO                    JM
028200      PFLAG% = K3%                                    JM
028210      IF IO% + 21 > LL% THEN GOSUB PRTLNE             JM
028220      IF PFLAG% = 0 THEN OUTO                          /* ERROR RETURN */ JM
028230      STR(OLINE$,IO%,21) = "<*& DOUBLE NEGATIVE >"    JM
028240      IO% = IO% + 21 :DNFL$(IW%)=" "                  JM
028440                                                    JM
028450      /* CHECK PREPOSITION FLAG */                     JM
028460                                                    JM
028470 PREPO: IF PREP_CNT$(<)"Y" THEN FUTUREO              JM
028480      IF PREPFL$(I%) = 0 THEN FUTUREO                  JM
028490      PFLAG% = K3%                                    JM
028500      IF IO% + 33 > LL% THEN GOSUB PRTLNE             JM
028510      IF PFLAG% = 0 THEN OUTO                          /* ERROR RETURN */ JM
028520      CONVERT PREPFL$(I%) TO OO$, PIC (###)           JM
028530      STR(OLINE$,IO%,33) = "<*& TOO MANY (" & OO$ & ") PREPOSI: JM
028540 TIONS *>"                                           JM
028550      PREPFL$(I%)=0                                    /* RESET PREP. FLAG */ JM
028560      IO% = IO% + 33                                    JM
028570                                                    JM
028580      /* LEAVE ROOM FOR FUTURE EXPANSION */             JM
028590                                                    JM
028600 FUTUREO: IO% = IO% + 1                                JM
028610                                                    JM
028620      /* NOW CHECK UNIQUE FLAGS - CHECK LONG PARAGRAPH FLAG */ JM
028630                                                    JM
028640      IF END_O_PAR$ <> "Y" THEN RESETO                /* NO END_O_PAR */ JM
028650      END_O_PAR$ = " "                                  /* RESET END_O_PAR FLAG */ JM
028660      IF LNG_PAR$(<)"Y" THEN SPRSTO                    /* PRINT THE PARAGRAPH */ JM
028670      IF SP% < LNGSNT% AND WP% < LNGWDS%                ! JM
028680      THEN SPRSTO                                        /* NOT TOO LONG */ JM
028690      PFLAG% = K3%                                    JM
028700      IF IO% + 57 > LL% THEN GOSUB PRTLNE             JM
028710      IF PFLAG% = 0 THEN OUTO                          JM
028720      CONVERT SP% TO OO$, PIC (###)                    JM

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 13:49 12/14/84 Page 50  
Input File is ANALYSIS in Library CRES@SRC on Volume CRES01

```

028730      CONVERT WP% TO OI$, PIC (####)           JM
028740      STR(OLINE$,IO%,57)="-< PARAGRAPH TOO LONG (" & OO$ & : JM
028750 " SENTENCES WITH "& OI$ & " WORDS ) >"           JM
028760      IO% = IO% + 57      :SP%,WP%=0           JM
028770      GOTO RESETO           JM
028780 SPRSTO: SP% = 0      /* CODE TO PUT END O_PARS ON SEPERATE LINES */ JM
028790      WP% = 0      /* WORDS PER PARAGRAPH = 0 */ JM
028800      PFLAG% = K3%           JM
028810      IF IO% > 1 THEN GOSUB PRTLNE           JM
028820      IF PFLAG% = 0 THEN OUTO           JM
028830           JM
028840      /* RESET WORK AREAS */           JM
028850           JM
028860 RESETO: IF IW% <= 1 THEN SINGLEO           JM
028870      FOR I% = 1 TO (IW%-1)           JM
028880          PASSVF$(I%) = PASSVF$(I%+1)           JM
028890          DNFL$(I%) = DNFL$(I%+1)           JM
028900          NOUNFL$(I%) = NOUNFL$(I%+1)           JM
028910          UCWFLG$(I%) = UCWFLG$(I%+1)           JM
028920          LCLSFL$(I%) = LCLSFL$(I%+1)           JM
028930          LSENT$(I%) = LSENT$(I%+1)           JM
028940          PREPFL$(I%) = PREPFL$(I%+1)           JM
028950          LW$(I%) = LW$(I%+1)           JM
028960      NEXT I%           JM
028970      PASSVF$(IW%) = " "           JM
028980      DNFL$(IW%) = " "           JM
028990      NOUNFL$(IW%) = " "           JM
029000      UCWFLG$(IW%) = " "           JM
029010      LCLSFL$(IW%) = 0           JM
029020      LSENT$(IW%) = 0           JM
029030      PREPFL$(IW%) = 0           JM
029040      LW$(IW%) = 0           JM
029050      GOTO POSTPONO           JM
029060 SINGLEO:           JM
029070      PASSVF$(1) = " "           JM
029080      DNFL$(1) = " "           JM
029090      NOUNFL$(1) = " "           JM
029100      UCWFLG$(1) = " "           JM
029110      LCLSFL$(1) = 0           JM
029120      LSENT$(1) = 0           JM
029130      PREPFL$(1) = 0           JM
029140      LW$(1) = 0           JM
029150 POSTPONO:           JM
029160      IF IW% > 0 THEN IW% = IW%-1           JM
029170      IF D% = 3 THEN OUTO           JM
029180           JM
029190      /* CHECK ON UNCOMMON WORDS AND PRINT IF TABLE IS FULL */ JM
029200           JM
029210      IF IDX% < 200 THEN OUTO           JM
029220      GOSUB UWL           JM
029221      GOSUB TOP           JM
029230 OUTO: IF CW$="Y" THEN CW$=" "           JM
029240      IF WCL%=0 THEN LIST%, CONSI% = 0           JM
029250      RETURN           /* PRT_WDS SUB. RETURN */ JM
029260           JM
029270 /****** JM
029280 /* JM

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 13:49 12/14/84 Page 51  
Input File is ANALYSIS in Library CRES@SRC on Volume CRES01

```

029290 /*      SUBROUTINE TTL_PGE SETS UP THE HEADER BLOCK FOR THE      */ JM
029300 /* ANALYSIS.  THIS BLOCK SHOWS WHAT FILE WAS SELECTED FOR THE */ JM
029310 /* ANALYSIS, THE TECHNICAL WORD LISTS USED, AND THE OPTIONS */ JM
029320 /* SELECTED. */ JM
029330 /* */ JM
029340 /****** */ JM
029350 */ JM
029360 TTL_PGE: /* SUBROUTINE TTL_PGE BEGINS */ JM
029370      C$=HEX(8001) /* TOP OF FORM */ JM
029380      WRITE #12, USING DOG, C$ JM
029390      C$ = HEX(0010) /* SKIP 10 LINES */ JM
029400      WRITE #12, USING DOG, C$ JM
029410      C$ = HEX(0001) JM
029420      FOR IQ%=1 TO IE% JM
029430      WRITE #12, USING DOG1, C$, BUFFA$(IQ%) JM
029440      NEXT IQ% JM
029450 RETURN JM
029460 DOG:  ### JM
029470 JM
029480 DOG1:  ### #####! JM
029490 ##### JM
029500 ENDSUB: RETURN /* ANALYSIS SUBROUTINE RETURN */ JM

```

Technical Report 86-001

THIS PAGE INTENTIONALLY LEFT BLANK.

Title Page Set-Up (TTLPGE)

TTLPGE in library CRES@SRC is the BASIC language program which sets up the header page of the CRES Analysis report. This header page is printed at the beginning of the analysis printout and includes the names of the document analyzed, any technical word lists utilized, and the options selected for inclusion in the analysis. A header page is included in the sample CRES output found in appendix D. The object language version of this program is TTLPGE in library CRES@SAV.

A complete program listing follows.

Technical Report 86-001

THIS PAGE INTENTIONALLY LEFT BLANK.

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 13:17 09/17/84 Page 1  
Input File is TITLPAGE in Library CRES@SRC on Volume CRES00

```

000010 /***** JM
000020 /* JM
000030 /* SUBROUTINE TTLPGE SETS UP THE HEADER BLOCK FOR THE RT
000040 /* ANALYSIS. THIS BLOCK SHOWS WHAT FILE WAS SELECTED FOR THE JM
000050 /* ANALYSIS, THE TECHNICAL WORD LISTS USED, AND THE OPTIONS JM
000060 /* SELECTED. JM
000070 /* JM
000080 /***** JM
000090 JM
000100 SUB "TTLPGE" ADDR(BUFFER$,PAG_TITL$,DOCFIL$,LIB$,DOCVOL$,RGL$, : JM
000110 PREP_CNT$,PASSV$,LNG_SEN$,LNG_PAR$,LNGWDS$,LNGSNT$,NEG$, : JM
000120 UNCOM$,NUM_SUBZ$,INT_ANL$,NP$,ADVFN$,PERS_SENT$,LNG_CLS$, : JM
000130 TRNK_PV$,TECFIL$,TECLIB$,TECVOL$,IE%) JM
000140 DIM PAG_TITL$60,DOCFIL$04,LIB$01,DOCVOL$6,RGL$1,PREP_CNT$1 JM
000150 DIM PASSV$1,LNG_SEN$1,LNG_PAR$1,LNGWDS$3,LNGSNT$2,NEG$1,UNCOM$1 JM
000160 DIM NUM_SUBZ$1,INT_ANL$1,NP$2,ADVFN$1,PERS_SENT$1,LNG_CLS$1 JM
000170 DIM TRNK_PV$1,TECFIL$(7)08,TECLIB$(7)08,TECVOL$(7)06 JM
000180 DIM BUFFER$(40)77,DAT$8,D$08,T$05 JM
000190 DAT$= DATE /* FETCH THE DATE */ JM
000200 D$= STR(DAT$,3,2)&"/"&STR(DAT$,5,2)&"/"&STR(DAT$,1,2) JM
000210 T$= STR(TIME,1,2)&":"&STR(TIME,3,2) JM
000220 BUFFER$(1)="!-----!" JM
000230 -----!" JM
000240 BUFFER$(2)="! : JM
000250 ! " JM
000260 BUFFER$(3)="! "&D$&" : JM
000270 "&T$&" : JM
000280 BUFFER$(4)="! ***** CRES ANALYSIS : JM
000290 ***** ! JM
000300 BUFFER$(5)=BUFFER$(2) JM
000310 BUFFER$(6)=BUFFER$(1) JM
000320 BUFFER$(7)=BUFFER$(2) JM
000330 IE%=8 JM
000340 IF PAG_TITL$= " " THEN GERRY JM
000350 BUFFER$(8)="! TITLE: "&STR(PAG_TITL$,1,60)&" ! " JM
000360 BUFFER$(9)=BUFFER$(2) JM
000370 IE%=10 JM
000380 GERRY: BUFFER$(IE%)="! INPUT FILE: "&DOCFIL$&" INPUT LI: JM
000390 BRARY: "&LIB$&" INPUT VOLUME: "&DOCVOL$&" ! " JM
000400 IE%=IE%+1 JM
000410 BUFFER$(IE%)=BUFFER$(2) /*SKIP A LINE */ JM
000420 IE%=IE%+1 JM
000430 BUFFER$(IE%)="! OPTIONS SELECTED FOR THIS : JM
000440 ANALYSIS: ! " JM
000450 IE%=IE%+1 JM
000460 BUFFER$(IE%)="! =====! JM
000470 ===== ! " JM
000480 IE%=IE%+1 JM
000490 BUFFER$(IE%)=BUFFER$(2) /*SKIP A LINE */ JM
000500 IE%=IE%+1 JM
000510 BUFFER$(IE%)="! REPORT FINAL READING GRADE LEVEL: "& : JM
000520 RGL$&" COUNT PREPOSITIONS: "&PREP_CNT$&" ! " RT
000530 IE%=IE%+1 JM
000540 BUFFER$(IE%)="! FLAG PASSIVE VOICE: "&PASSV$&" : JM
000550 FLAG LONG SENTENCES: "&LNG_SEN$&" ! " RT
000560 IE%=IE%+1 JM
000570 BUFFER$(IE%)="! FLAG DOUBLE NEGATIVES: "&NEG$&" : JM

```

# Technical Report 86-001

Wang VS Integrated Editor - Version 6.10.00 13:17 09/17/84 Page 2  
Input File is TITLPAGE in Library CRES@SRC on Volume CRES00

```

000580      FLAG UNCOMMON WORDS: " & UNCOM$ & "      ! "      RT
000590      IE%=IE%+1      JM
000600      BUFFER$(IE%)="! INCLUDE SUBSTITUTES: "& NUM_SUBZ$ & "      ! JM
000610      INTERIM ANALYSIS: "& INT_ANL$ & "      ! "      RT
000620      IE%=IE%+1      JM
000630      BUFFER$(IE%)="! NUMBER OF PAGES BETWEEN REPORTS: "&      ! JM
000640 STR(NP$,1,2) & " CHECK PERSONAL SENTENCES: " & PERS_SENT$ & " ! " RT
000650      IE%=IE%+1      JM
000700      BUFFER$(IE%)="! CHECK FOR TRUNCATED PASSIVE VOICE: " &      ! JM
000710 TRNK_PV$ & "      ! "      JM
000720      IE%=IE%+1      JM
000730      IF LNG_PAR$ = "Y" THEN BUFFER$(IE%)=BUFFER$(2)      RT
000740      IF LNG_PAR$ = "Y" THEN IE%=IE%+1
000750      BUFFER$(IE%)="! FLAG LONG PARAGRAPHS: "& LNG_PAR$ & "      ! RT
000760      ! "      JM
000770      IE%=IE%+1      JM
000780      IF LNG_PAR$="Y" THEN BUFFER$(IE%)="! HOW LONG?      ! JM
000790 WORDS = "& STR(LNGWDS$,1,3) & " SENTENCES = " &      ! JM
000800 STR(LNGSNT$,1,2) & "      ! "      JM
000810      IF LNG_PAR$="Y" THEN IE%=IE%+1      JM
000820      BUFFER$(IE%)=BUFFER$(2)      RT
000830      IE%=IE%+1      JM
000840      ZBT%=0      JM
000850      FOR ID%=1 TO 7      JM
000860      IF TECFIL$(ID%)=" " AND TECLIB$(ID%)=" " AND ! JM
000870      TECVOL$(ID%)=" " THEN NIL      JM
000880      IF ZBT%=1 THEN ZOUND      JM
000890      BUFFER$(IE%)="! TECHNICAL WORD LISTS : JM
000900 USED:      ! "      JM
000910      IE%=IE%+1      JM
000920      BUFFER$(IE%)="! =====! JM
000930 =====      ! "      JM
000940      IE%=IE%+1      JM
000950      BUFFER$(IE%)=BUFFER$(2)      JM
000960      IE%=IE%+1      JM
000970      ZBT%=1      JM
000980 ZOUND:      BUFFER$(IE%) = "! FILE: "& STR(TECFIL$(ID%),1,8) & ! JM
000990 " LIBRARY: " & STR(TECLIB$(ID%),1,8) & " VOLUME: " & ! JM
001000 STR(TECVOL$(ID%),1,6) & "      ! "      JM
001010      IE%=IE%+1      JM
001020 NIL:      NEXT ID%      JM
001030      BUFFER$(IE%)=BUFFER$(2)      JM
001040      IE%=IE%+1      JM
001050      BUFFER$(IE%)=BUFFER$(1)      JM
001060      END      JM

```

## COPY WORD LISTS TO TAPE SUBSYSTEM

### Copy Word Lists To Tape (CRESTAPE)

CRESTAPE in library CRES@SRC is the BASIC language program which copies one or more of the CRES word lists to tape. The object language version is CRESTAPE in library CRES@SAV.

This program requires the user to specify the word list to be copied and the name or number of the tape where the list is to be written. If the word list does not exist, or if there is no available tape drive, an appropriate message is displayed on the screen. If the word list does exist and there is an available drive, but no tape mounted, the user will be prompted to mount a tape. If a tape is already mounted, the program will copy the specified word list to the tape. On completion of the copy, the program returns to the word list specification screen to enable the user to write another word list to tape or to exit the Copy Word Lists to Tape option. The program requires that the word lists to be copied be specified one at a time. Each given list is copied successively to tape, not overwriting the previous word list, provided they are done in the same session. However, if the program is exited and then re-entered, the tape will be regarded as blank and word lists will be written starting at the beginning of the tape, writing over any previous word lists or other data.

Figure 5 presents a flowchart of the program to copy word lists to tape. A complete program listing follows.

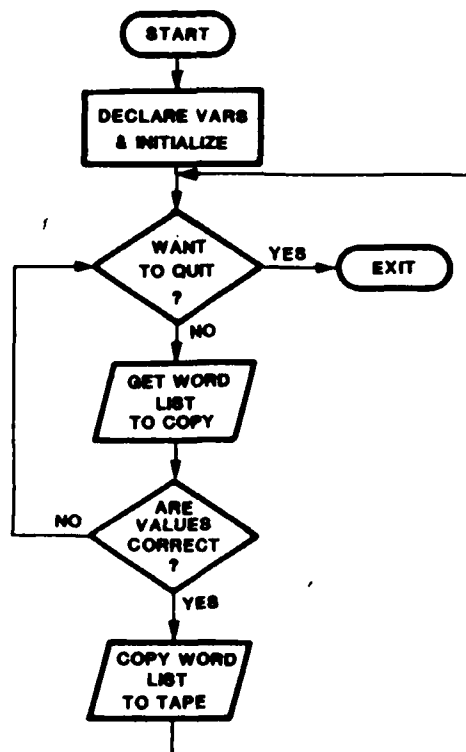


Figure 5. CRES Copy Word Lists to Tape Flowchart.

# Technical Report 86-001

\*\*\* Wang VS File Display Utility - Version 3.08.01 \*\*\* 01/13/86 13:42 Page 1  
Consecutive File CRESTAPE in Library CRES@SRC on Volume CRES01

```

000020/*****/RT
000040/* */RT
000060/* PROGRAM: BULDTAPE DATE REVISED: 11/25/83 BY: RT */RT
000080/* */RT
000100/* THIS PROGRAM READS THE DISK FILE AND PUTS THE DATA ON TAPE. */RT
000120/* */RT
000140/*****/RT
000160 RT
000180 DIM DISK BUF4$220,DISK BUF1$50,DISK BUF2$80,DISK BUF3$190 JM
000200 DIM TAPEINFO$255, FINDTAPE$24, TVOL$6, OPEN%(2), RET$5 RT
000220 RT
000240/* This is the CRES TAPE BUILDING SUBSYSTEM PROGRAM */ RT
000260/* */ RT
000280/* AUTHOR: Rose Tank */ RT
000300/* Date: November 25, 1983 */ RT
000320/* */ RT
000340/* This program copies a disk data file on to a tape for */ RT
000360/* distribution to the CRES users. */ RT
000380 RT
000400DIM DFILE$8,DLIB$8,DVOL$6,USER$3 RT
000420DIM HEADER$27,CODE$1,L$1,V$1,B$1,C$1,D$1,E$1,ER$40,F$40 RT
000440F$= "Filename not found on (library/volume)." RT
000460SEQNO%=0 RT
000480CALL "EXTRACT" ADDR("IV",DVOL$,"ID",USER$,"IL",DLIB$) RT
000500 RT
000520HEADER$ = "CRES DISK TO TAPE SUBSYSTEM" RT
000540NEXTFILE: RT
000560 DFILE$= " ":G$=HEX(81) RT
000580 ER$= " ":L$=HEX(81) RT
000600 V$=HEX(81): E$=HEX(8D) RT
000620FILMNU: RT
000640DIM ENTER$ 05 RT
000660DIM PF16$ 04 RT
000680ENTER$ = "ENTER" RT
000700PF16$ = "PF16" RT
000720ACCEPT !RT
000740 AT (01,01), !JM
000760"CRES: Disk to Tape (Build Tape Data File S !JM
000780election Screen)", !JM
000800 AT (02,01), !JM
000820"===== !RT
000840===== !JM
000860 AT (04,21), !RT
000880>Welcome to the", !RT
000900 AT (04,36), FAC(HEX(85)), HEADER$ , CH(27), !RT
000920 AT (07,02), !RT
000940" This Tape Program allows you to copy the word lists to tape fr !JM
000960om the disk.", !JM
000980 AT (08,02), !RT
001000" Please specify below the list you want to copy and the Tape Na !JM
001020me or Number.", !JM
001140 AT (13,30), !JM

```

# Technical Report 86-001

\*\*\* Wang VS File Display Utility - Version 3.08.01 \*\*\* 01/13/86 13:42 Page 2  
Consecutive File CRESTAPE in Library CRES@SRC on Volume CRES01

```

001160"File      =",                               :RT
001180      AT (13,44), FAC(HEX(81)),                DFILE$      , CH(08), :JM
001200      AT (14,30),                               :JM
001220"Library   =",                               :RT
001240      AT (14,44), FAC(HEX(81)),                DLIB$      , CH(08), :JM
001260      AT (15,30),                               :JM
001280"Volume    =",                               :RT
001300      AT (15,44), FAC(HEX(81)),                DVOL$      , CH(06), :JM
001320      AT (17,27),                               :JM
001340"Tape Name or Number =",                     :RT
001360      AT (17,49), FAC(HEX(81)),                TVOL$      , CH(06), :JM
001380      AT (19,17), FAC(E$),                     ER$        , CH(50), :RT
001400      AT (22,29),                               :JM
001420" Press",                                     :RT
001440      AT (22,36), FAC(HEX(85)),                ENTER$     , CH(05), :JM
001460      AT (22,42),                               :JM
001480"to continue",                               :RT
001500      AT (23,22),                               :RT
001520"Press",                                     :RT
001540      AT (23,28), FAC(HEX(84)),                PF16$     , CH(04), :RT
001560      AT (23,33),                               :RT
001580"to return to the Master Menu",               :RT
001600KEYS(BIN(0)&BIN(16)&BIN(32)),                   :RT
001620ON (BIN(16)&BIN(32)) GOTO OUT,OUT              :RT
001640                                              :RT
001660      SELECT #1,"DISKFILE",VARC,INDEXED,RECSIZE=50 , :RT
001680      KEYPOS=1, KEYLEN=20, IOERR GOTO IOERR1      :RT
001700      SELECT #2,"DISKFILE",VARC,INDEXED,RECSIZE=80 , :RT
001720      KEYPOS=1, KEYLEN=20, IOERR GOTO IOERR1      :RT
001740      SELECT #3,"DISKFILE",VARC,INDEXED,RECSIZE=195, :JM
001760      KEYPOS=1, KEYLEN=20, IOERR GOTO IOERR1      :RT
001780      SELECT #4,"DISKFILE",VARC,INDEXED,RECSIZE=220, :JM
001800      KEYPOS=1, KEYLEN=20, IOERR GOTO IOERR1      :JM
001820                                              :RT
001840/* SEE IF THE SPECIFIED FILE EXISTS */          :RT
001860                                              :RT
001880CALL "LIMITS"(DFILE$,DLIB$,DVOL$,CODE$,ER$)     :RT
001900IF CODE$="Y" THEN ZZZ                          :RT
001920IF CODE$="N" THEN GOSUB'7(F$,HEX(91),HEX(81),HEX(81)) :RT
001940IF CODE$="V" THEN GOSUB'7(STR(ER$,3),HEX(81),HEX(81),HEX(91)) :RT
001960IF CODE$="L" THEN GOSUB'7(STR(ER$,3),HEX(81),HEX(91),HEX(81)) :RT
001980IF CODE$="E" THEN GOSUB'7(STR(ER$,3),HEX(81),HEX(81),HEX(81)) :RT
002000GOTO FILMNU                                     :RT
002020                                              :RT
002040ER$=" " :G$,L$,V$,E$=HEX(8D)                   :RT
002060ZZZ: CALL "READFDR" ADDR(DFILE$,DLIB$,DVOL$,0%,"RS",RSIZ%,RET%) :RT
002080                                              :RT
002100      IF RSIZ%<51 THEN K=1                          :RT
002120      IF RSIZ%>50 AND RSIZ%<81 THEN K=2             :RT
002140      IF RSIZ%>80 AND RSIZ%<191 THEN K=3            :JM
002160      IF RSIZ%>190 THEN K=4                         :JM
002180                                              :RT

```

# Technical Report 86-001

\*\*\* Wang VS File Display Utility - Version 3.08.01 \*\*\* 01/13/86 13:42 Page 3  
Consecutive File CRESTAPE in Library CRES@SRC on Volume CRES01

```

002200      SELECT #5,"TAPEFIL1",TAPE,NL,RECSIZE=50 ,BLKSIZE=4000,      :JM
002220      DENSITY=1600,IOERR GOTO IOERR2                               RT
002240      :RT
002260      SELECT #6,"TAPEFIL2",TAPE,NL,RECSIZE=80 ,BLKSIZE=4000,      :JM
002280      DENSITY=1600,IOERR GOTO IOERR2                               RT
002300      :RT
002320      SELECT #7,"TAPEFIL3",TAPE,NL,RECSIZE=190,BLKSIZE=3800,      :JM
002340      DENSITY=1600,IOERR GOTO IOERR2                               RT
002360      :JM
002380      SELECT #8,"TAPEFIL4",TAPE,NL,RECSIZE=220,BLKSIZE=4400,      :JM
002400      DENSITY=1600,IOERR GOTO IOERR2                               JM
002420      :RT
002440      SEQNO%=SEQNO%+1                                             RT
002460/* 'extract' volume name and number of tape drives */         RT
002480      TAPEINFO$=HEX(02FD)                                         RT
002500      CALL "EXTRACT" ADDR("IV",DVOL$,"DL",TAPEINFO$)           RT
002520      :RT
002540      I=VAL(STR(TAPEINFO$,2))                                       RT
002560      DEVNO%=0                                                     RT
002580      SAVENO%=0                                                    RT
002600      FOR J=1 TO 1                                                 RT
002620          FINDTAPE$=STR(TAPEINFO$,2+J,1)                          RT
002640          CALL "EXTRACT" ADDR("D ",FINDTAPE$)                    RT
002660          IF STR(FINDTAPE$,9,6)=" " AND SAVENO%=0 THEN             :RT
002680              SAVENO%=VAL(STR(TAPEINFO$,2+J,1))                  RT
002700          IF STR(FINDTAPE$,9,6)<>TVOL$ THEN 2760                    JM
002720          DEVNO%=VAL(STR(TAPEINFO$,2+J,1))                         RT
002740          J=1                                                         RT
002760      NEXT J                                                         RT
002780      IF DEVNO%<>0 THEN 3080                                         JM
002800      :RT
002820      IF SAVENO%<>0 THEN 2920                                         JM
002840PRINT AT(22,1,);BELL;"All your tape drives are busy OR incorrec:RT
002860t Tape Name (or Number).": INPUT "PRESS ENTER ",RET$           RT
002880      GOTO CLSFL                                                    RT
002900      :RT
002920      DEVNO%=SAVENO%                                                 RT
002940      CALL "MOUNT" ADDR(SAVENO%,TVOL$,"N","E","R","N","B"," ",R%)   RT
002960      IF R%=0 THEN 3080                                              JM
002980      PRINT AT(22,1,);BELL;"Tape Mount failed ";R%                RT
003000      PRINT AT(23,1);"Touch 'ENTER' to return to menu. ";         RT
003020      INPUT RET$                                                    RT
003040      GOTO CLSFL                                                    RT
003060      :RT
003080      IF OPEN%(1) = 1 THEN CLOSE #K                                  RT
003100      OPEN NOGETPARM #K, INPUT, FILE=DFILE$, LIBRARY=DLIB$,         :RT
003120          VOLUME=DVOL$                                               RT
003140      OPEN%(1) = 1                                                  RT
003160      :RT
003180      IF OPEN%(2) = 1 THEN CLOSE #K+4                                JM
003200      OPEN NOGETPARM #K+4, OUTPUT, VOLUME=TVOL$, FILESEQ=SEQNO%     JM
003220      OPEN%(2) = 1                                                  RT

```

# Technical Report 86-001

\*\*\* Wang VS File Display Utility - Version 3.08.01 \*\*\* 01/13/86 13:42 Page 4  
Consecutive File CRESTAPE in Library CRES@SRC on Volume CRES01

```

003240                                     RT
003260DISPLAY                             !RT
003280             AT (01,03),                 !RT
003300             "CRES Subsystem: Build Tape Data File", !RT
003320             "                               (Writing to Tape)", !RT
003340             AT (02,03),                 !RT
003360"===== !RT
003380"=====, !RT
003400             AT(09,21), !RT
003420             "Writing to the following Tape : ", !RT
003440             AT (11,25), !RT
003460             " VOLUME ==> ",TVOL$,"" !RT
003480                                     RT
003500 ON(K) GOTO 3520,3540,3560,3570 !RT
003520 READ #K,DISK_BUF1$, EOD GOTO CLSFL :GOTO 3580 !RT
003540 READ #K,DISK_BUF2$, EOD GOTO CLSFL :GOTO 3580 !RT
003560 READ #K,DISK_BUF3$, EOD GOTO CLSFL :GOTO 3580 !RT
003570 READ #K,DISK_BUF4$, EOD GOTO CLSFL :GOTO 3580 !RT
003580 ON(K) GOTO 3600,3620,3640,3660 !RT
003600 WRITE #K+4, DISK_BUF1$, DATA GOTO IOERR2 :GOTO 3500 !RT
003620 WRITE #K+4, DISK_BUF2$, DATA GOTO IOERR2 :GOTO 3500 !RT
003640 WRITE #K+4, DISK_BUF3$, DATA GOTO IOERR2 :GOTO 3500 !RT
003660 WRITE #K+4, DISK_BUF4$, DATA GOTO IOERR2 :GOTO 3500 !RT
003680                                     RT
003700CLSFL: IF OPEN%(1) = 1 THEN CLOSE #K !RT
003720 IF OPEN%(2) = 1 THEN CLOSE #K+4 !RT
003740 MAT OPEN%=ZER !RT
003760 GOTO NEXTFILE !RT
003780                                     RT
003800OUT: CALL "DISMOUNT" ADDR(TVOL$,"T",R%) !RT
003820 END !RT
003840                                     RT
003860/* THIS FUNCTION PRINTS OUT ERROR MESSAGES AND */ !RT
003880/* SETS UP FLASHING FAC'S IN APPROPRIATE PLACES */ !RT
003900DEF FN'7(ER$,B$,C$,D$) !RT
003920 PRINT !RT
003940 E$=HEX(94) !RT
003960 G$=B$ !RT
003980 L$=C$ !RT
004000 V$=D$ !RT
004020RETURN !RT
004040                                     RT
004060IOERR1: PRINT PAGE, AT(01,01), "I/O ERROR ON DISK DATA FILE #1. !RT
004080FILE STATUS IS : ";FS(#1) !RT
004100 STOP " PRESS ENTER TO RETURN TO PREVIOUS MENU" !RT
004120 GOTO CLSFL !RT
004140                                     RT
004160IOERR2: PRINT PAGE, AT(01,01), "I/O ERROR ON TAPE DATA FILE #2. !RT
004180FILE STATUS IS : ";FS(#K) !RT
004200 STOP " PRESS ENTER TO RETURN TO PREVIOUS MENU" !RT
004220 GOTO CLSFL !RT
*** End of Listing *****

```

REFERENCES

A need to address illiteracy problems in the military services.  
Washington, DC: General Accounting Office, 1977.

Improved management of maintenance manuals needed in DOD.  
Washington, DC: General Accounting Office, 1979.

Chief, Naval Education and Training, Code N-5. Letter to Commander, Naval Training Equipment Center, Training Analysis and Evaluation Group (Code 1), 29 June 1978.

Military Specification. Manuals, Technical: General style and format requirements. (MIL-M-38784B), Washington, DC: Department of Defense, 16 April 1983.

Kincaid, J. P., Aagard, J. A., & O'Hara, J. W. Development and test of a Computer Readability Editing System (CRES). (Technical Report No. 83). Orlando, FL: Training Analysis and Evaluation Group, 1980.

Kincaid, J. P., McDaniel, W. C., & Pierce, M. Word lists to simplify vocabulary of technical information. (Technical Report No. 164). Orlando, FL: Naval Training Equipment Center, 1985.

Schalow, S. R., & Mears, J. E. Computer readability and editing system (CRES): User's reference manual. (Technical Report 86-xxx). Orlando, FL: Naval Training Systems Center, in press.

Kincaid, J. P., & Pierce, M. Computer readability editing system (CRES) for writers and editors. (Technical Report 86-xxx). Orlando, FL: Naval Training Systems Center, in press.

Technical Report 86-001

THIS PAGE INTENTIONALLY LEFT BLANK.

Technical Report 86-001

APPENDIX A

Technical Report 86-001

THIS PAGE INTENTIONALLY LEFT BLANK.

## APPENDIX A

The following provides definitions of the CRES word lists, and the control file for each. Other lists can be created with different names and used in the CRES analysis, but the format for each would be identical to those listed below.

CMNWDLST	This is the Common Word List used in the CRES analysis.
WRDSBLST	This is the Word Substitute List used in the CRES analysis.
PHRSBLST	This is the Phrase Substitute List used in the CRES analysis.
TECWDLST	This is a Technical Word List used in the CRES analysis. There can be any number of these lists created, each with a different name. Up to seven may be used in any given analysis.
ABACGLOS	This is the Abbreviation/Acronym Glossary used in the CRES analysis.
GLOSSARY	This is the Reference Glossary. The words contained on this list are not used in the CRES analysis.

# Technical Report 86-001

CONTROL FILE UTILITY PROGRAM  
DATE: 01/07/85 TIME: 10:42  
List of Control File

## Control File Characteristics

File Name: CMNWDLST

Records are fixed-length, record size is 0040

File organization is indexed, with key field = WORD

OPERATION	ALLOWED?
File report	YES
Record update	YES
Record deletion	YES

Description: THIS FORMAT IS FOR THE RECORDS OF THE COMMON WORD LIST  
ASSOCIATED WITH CRES

Field Name	Start Posn	Int Fmt	Int Len	Ext Len	Rep Len	Dec Pos	Occur Count	Report/ Update	Blank After	Date/ Time	Cum Field	Valid- ation	Alias Name
WORD	1	C	20	20	20	0	1	R/U	YES				WORD
PRTOFSPC	21	C	1	1	1	0	1	R/U	YES				PART OF SPEECH
PPART	22	C	1	1	1	0	1	R/U	YES				PAST PARTICIPLE
NEGATIVE	23	C	1	1	1	0	1	R/U	YES				NEGATIVE FLAG
TOBE	24	C	1	1	1	0	1	R/U	YES				TO BE FLAG
FILLER	25	C	1	1	1	0	1	R/U	YES				
ROOTWORD	26	C	1	1	1	0	1	R/U	YES				ROOT WORD FLAG
FILLER2	27	C	1	1	1	0	1	R/U	YES				
PROTECT	28	C	3	3	3	0	1	R/U	YES				PROTECTION BYTES
FILLER3	31	C	10	10	10	0	1	R/U	YES				

# Technical Report 86-001

CONTROL FILE UTILITY PROGRAM  
DATE: 01/07/85 TIME: 10:43  
List of Control File

## Control File Characteristics

File Name: WRDSBLST

Records are fixed-length, record size is 0080

File organization is indexed, with key field = WORD

OPERATION	ALLOWED?
File report	YES
Record update	YES
Record deletion	YES

Description: THIS IS THE RECORD FORMAT FOR THE WORD SUBSTITUTION LIST  
ASSOCIATED WITH CRES

Field Name	Start Posn	Int Fmt	Int Len	Ext Len	Rep Len	Dec Pos	Occur Count	Report/ Update	Blank After	Date/ Time	Cum Field	Valid- ation	Alias Name
WORD	1	C	20	20	20	0	1	R/U	YES				WORD
SUBONE	21	C	20	20	20	0	1	R/U	YES				FIRST SUBSTITUTE
SUBTWO	41	C	15	15	15	0	1	R/U	YES				SECOND SUBSTITUTE
PRTOSPC	56	C	1	1	1	0	1	R/U	YES				PART OF SPEECH
PPFL	57	C	1	1	1	0	1	R/U	YES				PAST PARTICIPLE
NEGFL	58	C	1	1	1	0	1	R/U	YES				NEGATIVE FLAG
PROTECT	59	C	3	3	3	0	1	R/U	YES				PROTECTION BYTES
FILLER	62	C	19	19	19	0	1	R/U	YES				

# Technical Report 86-001

CONTROL FILE UTILITY PROGRAM  
DATE: 01/07/85 TIME: 10:42  
List of Control File

## Control File Characteristics

File Name: PHRSBLST

Records are fixed-length, record size is 0070

File organization is indexed, with key field = WORD

OPERATION	ALLOWED?
File report	YES
Record update	YES
Record deletion	YES

Description: THIS IS THE RECORD FORMAT FOR THE PHRASE SUBSTITUTION LIST  
ASSOCIATED WITH CRES

Field Name	Start Posn	Int Fmt	Int Len	Ext Len	Rep Len	Dec Pos	Occur Count	Report/Update	Blank After	Date/Time	Cum Field	Valid-ation	Alias Name
WORD	1	C	24	24	24	0	1	R/U	YES				PHRASE
SUBONE	25	C	20	20	20	0	1	R/U	YES				FIRST SUBSTITUTE
SUBTWO	45	C	20	20	20	0	1	R/U	YES				SECOND SUBSTITUT
PROTECT	65	C	3	3	3	0	1	R/U	YES				PROTECTION BYTES
FILLER	68	C	3	3	3	0	1	R/U	YES				

# Technical Report 86-001

CONTROL FILE UTILITY PROGRAM

DATE: 01/07/85 TIME: 10:42

List of Control File

## Control File Characteristics

File Name: PHRSBLST

Records are fixed-length, record size is 0070

File organization is indexed, with key field = WORD

OPERATION	ALLOWED?
File report	YES
Record update	YES
Record deletion	YES

Description: THIS IS THE RECORD FORMAT FOR THE PHRASE SUBSTITUTION LIST  
ASSOCIATED WITH CRES

Field Name	Start Posn	Int Fmt	Int Len	Ext Len	Rep Len	Dec Pos	Occur Count	Report/Update	Blank After	Date/Time	Cum Field	Valid-ation	Alias Name
WORD	1	C	24	24	24	0	1	R/U	YES				PHRASE
SUBONE	25	C	20	20	20	0	1	R/U	YES				FIRST SUBSTITUTE
SUBTWO	45	C	20	20	20	0	1	R/U	YES				SECOND SUBSTITUTE
PROTECT	65	C	3	3	3	0	1	R/U	YES				PROTECTION BYTES
FILLER	68	C	3	3	3	0	1	R/U	YES				

# Technical Report 86-001

CONTROL FILE UTILITY PROGRAM  
DATE: 01/07/85 TIME: 10:43  
List of Control File

## Control File Characteristics

File Name: TECWDLST

Records are fixed-length, record size is 0040

File organization is indexed, with key field = WORD

## OPERATION ALLOWED?

File report	YES
Record update	YES
Record deletion	YES

Description: THIS IS THE RECORD FORMAT FOR THE TECHNICAL WORD LISTS  
ASSOCIATED WITH CRES

Field Name	Start Posn	Int Fmt	Int Len	Ext Len	Rep Len	Dec Pos	Occur Count	Report/ Update	Blank After	Date/ Time	Cum Field	Valid- ation	Alias Name
WORD	1	C	20	20	20	0	1	R/U	YES				WORD
PRTOFSPC	21	C	1	1	1	0	1	R/U	YES				PART OF SPEECH
PPART	22	C	1	1	1	0	1	R/U	YES				PAST PARTICIPLE
NEGATIVE	23	C	1	1	1	0	1	R/U	YES				NEGATIVE FLAG
TOBE	24	C	1	1	1	0	1	R/U	YES				FORM OF TO BE FLAG
FILLER	25	C	1	1	1	0	1	R/U	YES				
ROOTWORD	26	C	1	1	1	0	1	R/U	YES				ROOT WORD FLAG
MTWD	27	C	1	1	1	0	1	R/U	YES				MANDATORY TECHNICAL
PROTECT	28	C	3	3	3	0	1	R/U	YES				PROTECTION BYTES
FILLER2	31	C	10	10	10	0	1	R/U	YES				

# Technical Report 86-001

DATE: 01/07/85 TIME: 10:38  
List of Control File

## Control File Characteristics

File Name: ABACGLOS

Records are fixed-length, record size is 0190

File organization is indexed, with key field = WORD

OPERATION	ALLOWED?
File report	YES
Record update	YES
Record deletion	YES

Description: THIS IS THE RECORD FORMAT FOR THE ABBREVIATION/ACRONYM  
GLOSSARY ASSOCIATED WITH CRES

Field Name	Start Posn	Int Fmt	Int Len	Ext Len	Rep Len	Dec Pos	Occur Count	Report/ Update	Blank After	Date/ Time	Cum Field	Valid- ation	Alias Name
WORD	1	C	20	20	20	0	1	R/U	YES				ABBREV/ACRONYM
TYPE	21	C	1	1	1	0	1	R/U	YES				B-ABBREV,C-ACRONYM
DEF1	22	C	55	55	55	0	1	R/U	YES				DEFINITION LINE 1
DEF2	77	C	55	55	55	0	1	R/U	YES				DEFINITION LINE 2
DEF3	132	C	55	55	55	0	1	R/U	YES				DEFINITION LINE 3
FILLER	187	C	4	4	4	0	1	R/U	YES				

Technical Report 86-001

THIS PAGE INTENTIONALLY LEFT BLANK.

Technical Report 86-001

APPENDIX B

Technical Report 86-001

THIS PAGE INTENTIONALLY LEFT BLANK.

# Technical Report 86-001

## APPENDIX B

The following is a list of the included programs and data files, and the library in which they reside:

File Name	Library	Description
LIMITS	@SYSSAV@	Checks for the existence of specific files
FINDUSER	@SYSSAV@	Finds a specific user logged on the system
ABBRELST	CRES@BAK	Backup copy of Abbreviation/Acronym list
ADMCLER	CRES@BAK	Backup copy of Administrative/Clerical Technical Word List
AIREQUI	CRES@BAK	Backup copy of Aircraft Equipment Technical Word List
CMNWDLST	CRES@BAK	Backup copy of the Common Word List
COMMNAV	CRES@BAK	Backup copy of the Communication/Navigation Technical Word List
ELIMSPCH	CRES@BAK	Backup copy of a table of characters to be eliminated from documents before analysis
GENELCT	CRES@BAK	Backup copy of the General Electronics Technical Word List
GLOSSARY	CRES@BAK	Backup copy of the Glossary
HMELCT	CRES@BAK	Backup copy of the Hull, Mechanical, and Electrical Technical Word List
PHRSBLST	CRES@BAK	Backup copy of the Phrase Substitution List
PROPENG	CRES@BAK	Backup copy of the Propulsion Engineering Technical Word List
WRDSBLST	CRES@BAK	Backup copy of the Word Substitution List
WSCONT	CRES@BAK	Backup copy of the Weapon Systems Control Technical Word List
ABACGLOS	CRES@CTL	Control file for the Abbreviation/Acronym List
CMNWDLST	CRES@CTL	Control file for the Common Word List
GLOSSARY	CRES@CTL	Control file for the Glossary
PHRSBLST	CRES@CTL	Control file for the Phrase Substitution List
TECWDLST	CRES@CTL	Control file for the Technical Word Lists
WRDSBLST	CRES@CTL	Control file for the Word Substitution List
ABBRELST	CRES@DAT	Abbreviation/Acronym List
ADMCLER	CRES@DAT	Administrative/Clerical Technical Word List
AIREQUI	CRES@DAT	Aircraft Equipment Technical Word List
CMNWDLST	CRES@DAT	Common Word List
COMMNAV	CRES@DAT	Communications/Navigation Technical word List
ELIMSPCH	CRES@DAT	Table of characters to be eliminated from documents before analysis
GENELCT	CRES@DAT	General Electronics Technical Word List
GLOSSARY	CRES@DAT	Technical Word Glossary

# Technical Report 86-001

HMELCT	CRES@DAT	Hull, Mechanical, and Electrical Technical Word List
PHRSBLST	CRES@DAT	Phrase Substitution List
PROPENG	CRES@DAT	Propulsion Engineering Technical Word List
WRDSBLST	CRES@DAT	Word Substitution List
WSCONT	CRES@DAT	Weapon Systems Control Technical Word List
ANALDISP	CRES@PRC	Procedure Language routine to display completed analyses at the terminal
CRES	CRES@PRC	Procedure Language routine that serves as the main driver for the CRES system
CRES@BCG	CRES@PRC	Procedure Language routine that serves as a driver for the Maintain Glossaries subsystem
CRES@CMW	CRES@PRC	Procedure Language routine that serves as a driver for the Create/Maintain Common Word List subsystem
CRES@SWL	CRES@PRC	Procedure Language routine that serves as a driver for the Create/Maintain Technical Word List subsystem
CRES@WSL	CRES@PRC	Procedure Language routine that serves as a driver for the Create/Maintain Substitution Lists subsystem
DISPFILE	CRES@PRC	Procedure Language routine to display selected word lists at the terminal
ANALYSIS	CRES@SAV	The object code for the Analysis Program
CRES@BCG	CRES@SAV	The object code for the Maintain Abbreviation/ Acronym Program
CRES@BCP	CRES@SAV	The object code for the Print Abbreviation/ Acronym Program
CRES@CPA	CRES@SAV	The object code for the Print Common Words By Attribute Program
CRES@CWL	CRES@SAV	The object code for the Create/Maintain Common Word List Program
CRES@CWP	CRES@SAV	The object code for the Print Common Word List Program
CRES@ERW	CRES@SAV	The object code for the Expand Root Words Program
CRES@PSL	CRES@SAV	The object code for the Create/Maintain Phrase Substitution List Program
CRES@PSP	CRES@SAV	The object code for the Print Phrase Substitution List Program
CRES@RND	CRES@SAV	The object code for random number generation
CRES@SBP	CRES@SAV	The object code for the Print Combination of Word and Phrase Substitution Lists Program
CRES@TPA	CRES@SAV	The object code for the Print Technical Words by Attribute Program

Technical Report 86-001

CRES@TWL	CRES@SAV	The object code for the Create/Maintain Technical Word List Program
CRES@TWP	CRES@SAV	The object code for the Print Technical Word List Program
CRES@WSL	CRES@SAV	The object code for the Create/Maintain Word Substitution List Program
CRES@WSP	CRES@SAV	The object code for the Print Word Substitution List Program
CRESGLOP	CRES@SAV	The object code for the Glossary Print Program
CRESGLOS	CRES@SAV	The object code for the Maintain Glossary Program
CRESTAPE	CRES@SAV	The object code for the Word List to Tape Program
CRSEN1	CRES@SAV	The object code for a logoff graphics program

Technical Report 86-001

THIS PAGE INTENTIONALLY LEFT BLANK.

Technical Report 86-001

APPENDIX C

Technical Report 86-001

THIS PAGE INTENTIONALLY LEFT BLANK.

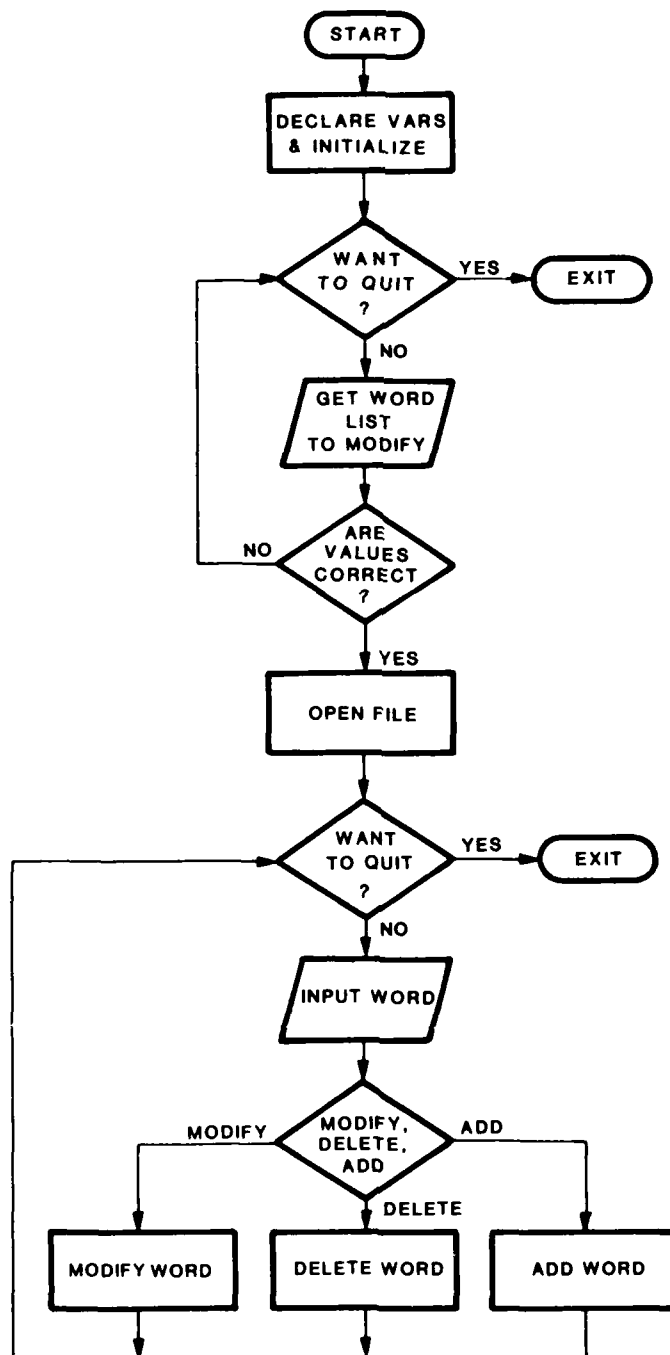
## APPENDIX C

There are three categories of programs in the CRES system that are essentially the same in all the subsystems. They are the programs to create and/or maintain a word list or glossary, the programs to print a word list or glossary, and the programs to print a word list or glossary according to a specified attribute or combination of attributes. This appendix presents the basic flowcharts for these three categories of programs.

Technical Report 86-001

THIS PAGE INTENTIONALLY LEFT BLANK.

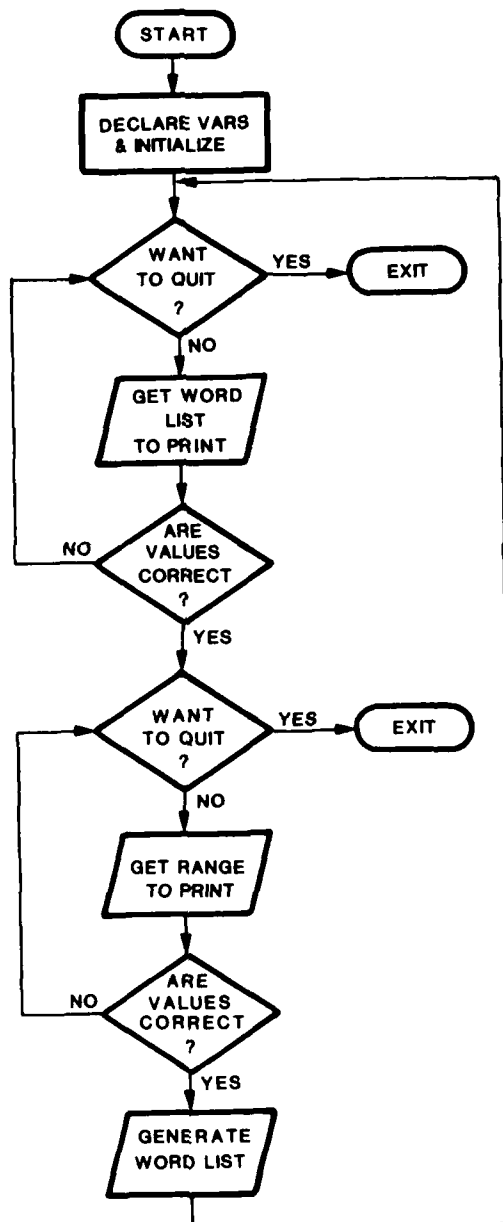
CRES Create/Maintain Word List Flowchart



Technical Report 86-001

THIS PAGE INTENTIONALLY LEFT BLANK.

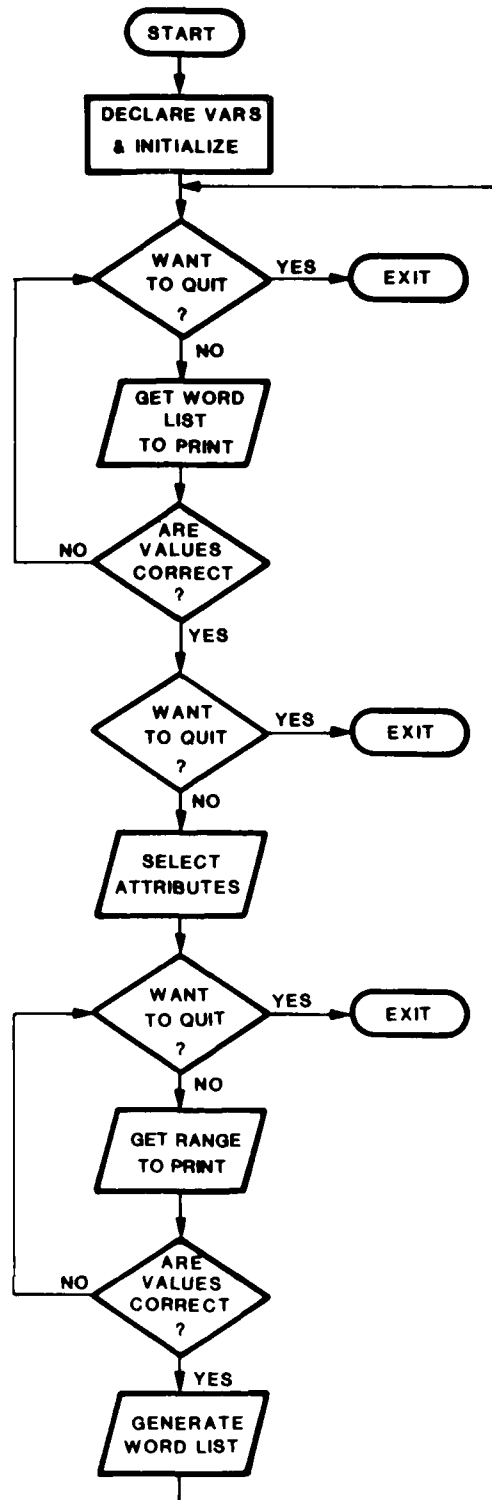
CRES Print Word List Flowchart



Technical Report 86-001

THIS PAGE INTENTIONALLY LEFT BLANK.

CRES Print Word List by Attribute Flowchart



Technical Report 86-001

THIS PAGE INTENTIONALLY LEFT BLANK.

Technical Report 86-001

APPENDIX D

Technical Report 86-001

THIS PAGE INTENTIONALLY LEFT BLANK.

APPENDIX D

Sample Text Input to CRES

The honors and ceremonies prescribed in these regulations may be dispensed with when directed by the Secretary of the Navy, or when requested by an individual to whom such honors and ceremonies are due. No salute shall be fired in honor of any nation or of any official of any nation not formally recognized by the Government of the United States; and, except as authorized by the Secretary of the Navy, no other honors or ceremonies prescribed in these regulations shall be rendered or exchanged with such nations or officials. Should the required number or frequency of international salutes, official visits, or other honors and ceremonies be deemed excessive, the senior officer present in the United States naval service may make alterations, subject to the requirements of international courtesy, such modifications as circumstances warrant and as may be agreed upon with responsible officials or the senior officer present of the nation involved. The National Anthem of the United States, "The Star Spangled Banner", when played by a naval band shall be played in its entirety as written and as prescribed in the official United States Navy Band arrangement which is designated as the official Department of Defense arrangement. The playing of the National Anthem of the United States, or of any other country, as a part of a medley is prohibited. When a foreign national anthem is prescribed in connection with honors, and it is considered appropriate to perform the National Anthem of the United States therewith, the National Anthem of the United States shall be performed last. On other occasions when a foreign national anthem (or anthems) is performed, the National Anthem of the United States will be performed last, except when performed in conjunction with Morning Colors.

Whenever the National Anthem is played, all naval service personnel not in formation shall stand at attention and face the national ensign; but in the event that the national ensign is not being displayed, they shall face the source of the music. When covered they shall come to the salute at the first note of the anthem; and shall remain at the salute until the last note of the anthem. Persons in formation are brought to order arms or called to attention as appropriate. The formation commander shall face in the direction of the ensign or in the absence of the ensign shall face in the direction of the music and shall render the appropriate salute for his unit. Persons in formation participating in a ceremony shall, on command, follow the procedure prescribed for the ceremony. Persons in vehicles or in boats shall follow the procedure prescribed for such persons during colors; persons in civilian clothes shall comply with the rules and customs established for civilians. The same marks of respect prescribed during the playing of the National Anthem shall be shown during the playing of a foreign national anthem. The

## Technical Report 86-001

ceremonial hoisting and lowering of the national ensign at 0800 and sunset at a naval command ashore or aboard a ship of the Navy not under way shall be known as Morning Colors and Evening Colors, respectively, and shall be carried out as prescribed in this article. The guard of the day and the bands shall be paraded in the vicinity of the point of hoist of the ensign.

"Attention" shall be sounded, followed by the playing of the National Anthem by the band. At Morning Colors, the ensign shall be started up at the beginning of the music and hoisted smartly to the peak or truck. At Evening Colors, the ensign shall be started from the peak or truck at the beginning of the music and the lowering so regulated as to be completed at the last note.

It is the responsibility of transferring activities to ensure that members acquire obligated service necessary for tour length selected by the member. If, at the time of transfer, a member is undecided as to which tour (accompanied/all others) he/she desires, sufficient obligated service to complete the "all others" tour for the area regardless of the member's pay grade or length of service, personnel ordered to an overseas activity (land-based or afloat) shall have sufficient obligated service to ensure completion of the appropriate accompanied tour prescribed for the area to which ordered prior to execution of orders. Attainment of obligated service through actual reenlistment/extension of enlistment is required. Page 13 entry for attainment of obligated service is prohibited. Personnel approaching fleet reserve/retirement eligibility. Personnel who have completed 15 or more years active duty for the period of time required to complete the applicable tour prescribed for the overseas area to which assigned. Transfer to the Fleet Reserve or Retired List will not normally be authorized prior to completion of the prescribed overseas tour. Personnel approaching retirement eligibility are cautioned of the fact that the requirement to complete the applicable tour for the overseas area may disallow their eligibility to request a twilight tour. Personnel (other than CT, OT, RM, and enlisted women in those ratings listed in Article 3.181) will not normally be involuntarily required to serve consecutive overseas tours unless Navy requirements (billet priorities) direct such action. Personnel assigned consecutive overseas tours as a result of voluntary action will be required to serve the appropriate accompanied/all others tour length.

## Technical Report 86-001

### CRES Analysis of Sample Text

The following pages contain the CRES analysis of the preceding text. The first page is a header, which is printed at the beginning of each analysis. This header contains all the information pertinent to the particular analysis. The date and time of the analysis, any title specified, the name and location of the file which was analyzed, and the options selected for the given analysis are all listed for future reference. Then the text is printed with editorial insertions flagging long sentences and long paragraphs, with the associated word or sentence count, double negatives, passive voice construction, and any other attributes which were indicated during option selection. Word and phrase substitutes are included in parentheses if that option was specified. At the conclusion of the text the Reading Grade Analysis is printed and the "uncommon" words encountered in the text are listed in alphabetical order.

Technical Report 86-001

06/04/85

\*\*\*\*\*

CRES ANALYSIS

\*\*\*\*\*

13:23

TITLE: CRES DEMO

INPUT FILE: 0001

INPUT LIBRARY: R

INPUT VOLUME: SYSVOL

OPTIONS SELECTED FOR THIS ANALYSIS:

=====

REPORT FINAL READING GRADE LEVEL: Y

FLAG PASSIVE VOICE: Y

FLAG DOUBLE NEGATIVES: Y

INCLUDE SUBSTITUTES: Y

NUMBER OF PAGES BETWEEN REPORTS: 05

CHECK FOR TRUNCATED PASSIVE VOICE: Y

COUNT PREPOSITIONS: Y

FLAG LONG SENTENCES: Y

FLAG UNCOMMON WORDS: Y

INTERIM ANALYSIS: Y

CHECK PERSONAL SENTENCES: Y

FLAG LONG PARAGRAPHS: Y

HOW LONG?

WORDS = 200

SENTENCES = 10

# Technical Report 86-001

PAGE 1

The honors and ceremonies prescribed in these regulations may be dispensed  
< PASSIVE VOICE > with when directed by the Secretary of the Navy, or when  
< requested > \* ASKED \* by an individual to whom such honors and ceremonies  
are due. < 34 \* > No salute shall be fired < PASSIVE VOICE \* > in honor of any  
nation or of any official of any nation not [formally] recognized by the  
Government of the United States; < 27 \* > \* TOO MANY ( 6) PREPOSITIONS \* >  
and, except as authorized by the Secretary of the Navy, no  
< DOUBLE NEGATIVE \* > other honors or ceremonies prescribed in these  
regulations shall be rendered < \* GAVE/GIVEN / MADE \* > < PASSIVE VOICE \* > or  
exchanged with such nations or officials. < 29 \* > Should the  
required < \* NEEDED \* > number or frequency of international salutes,  
official visits, or other honors and ceremonies be  
deemed < \* THOUGHT / CONSIDERED \* > < PASSIVE VOICE \* > excessive, the senior  
officer present in the United States naval service may make [alterations],  
subject to the requirements of international [courtesy], such  
modifications < \* CHANGES \* > as circumstances [warrant] and as may be agreed  
< PASSIVE VOICE \* > upon with responsible officials or the senior officer  
present of the nation involved. < 62 \* > < \* LIST? \* > The National Anthem of the  
United States, "The Star Spangled Banner", when played by a naval band shall  
be played < \* PASSIVE VOICE \* > in its [entirety] as written and as prescribed  
in the official United States Navy Band arrangement which is  
designated < \* SPECIFIED / CHOSEN \* > < PASSIVE VOICE \* > as the official  
Department of Defense arrangement. < 46 \* > The playing of the National Anthem  
of the United States, or of any other country, as a part of a [medley] is  
prohibited. < 23 \* > < \* PASSIVE VOICE \* > When a foreign national anthem is  
prescribed < PASSIVE VOICE \* > in connection with < \* WITH / ABOUT \* > honors,

# Technical Report 86-001

PAGE 2

and it is considered < \* PASSIVE VOICE \* > appropriate < \* PROPER / RIGHT \* > to perform the National Anthem of the United States therewith, < \* WITH \* > the National Anthem of the United States shall be performed < \* PASSIVE VOICE \* > last. < \* 37 \* > On other occasions when a foreign national anthem (or anthems) is performed, < \* PASSIVE VOICE \* > the National Anthem of the United States will be performed < \* PASSIVE VOICE \* > last, except when performed in conjunction with < \* WITH / FOR \* > Morning Colors. < \* 31 \* > Whenever the National Anthem is played, < \* PASSIVE VOICE \* > all naval service personnel not in formation shall stand at attention and face the national ensign; < \* 23 \* > but in the event that < \* IF \* > the national ensign is not being displayed, < \* PASSIVE VOICE \* > they shall face the source of the music. When covered they shall come to the salute at the first note of the anthem; < \* TOO MANY ( 3 ) PREPOSITIONS \* > and shall remain < \* STAY \* > at the salute until the last note of the anthem. < \* TOO MANY ( 3 ) PREPOSITIONS \* > Persons in formation are brought < \* PASSIVE VOICE \* > to order arms or called to attention as appropriate. < \* PROPER / RIGHT \* > < \* TOO MANY ( 3 ) PREPOSITIONS \* > The formation commander shall face in the direction of < \* DIRECTING \* > the ensign or in the absence of the ensign shall face in the direction of < \* DIRECTING \* > the music and shall render < \* GIVE / MAKE \* > the appropriate < \* PROPER / RIGHT \* > salute for his unit. < \* 35 \* > < \* TOO MANY ( 7 ) PREPOSITIONS \* > Persons in formation participating < \* TAKING PART \* > in a ceremony shall, on command, follow the procedure prescribed for the ceremony. < \* TOO MANY ( 4 ) PREPOSITIONS \* > Persons in vehicles or in boats shall follow the procedure prescribed for such persons during colors; < \* TOO MANY ( 4 ) PREPOSITIONS \* > persons in civilian clothes shall comply with the rules and customs

# Technical Report 86-001

PAGE 3

established < \* SET UP / PROVED \* > for civilians.

< \* TOO MANY ( 3) PREPOSITIONS \* > The same marks of respect prescribed during the playing of the National Anthem shall be shown < \* PASSIVE VOICE \* > during the playing of a foreign national anthem. < \* 24 \* >

< \* TOO MANY ( 5) PREPOSITIONS \* > The [ceremonial] hoisting and lowering of the national ensign at 0800 and sunset at a naval command ashore or aboard a ship of the Navy not under way shall be known < \* PASSIVE VOICE \* > as Morning Colors and Evening Colors, [respectively], and shall be carried

< \* PASSIVE VOICE \* > out as prescribed in this article. < \* 48 \* > The guard of the day and the bands shall be [paraded] in the vicinity of the the point of hoist of the ensign. < \* 23 \* > < \* TOO MANY ( 5) PREPOSITIONS \* > "Attention" shall be sounded, < \* PASSIVE VOICE \* > followed by the playing of the National Anthem by the band. < \* TOO MANY ( 3) PREPOSITIONS \* > At Morning Colors, the ensign shall be started < \* PASSIVE VOICE \* > up at the [beginning] of the music and hoisted smartly to the peak or truck. < \* 23 \* > At Evening Colors, the ensign shall be started < \* PASSIVE VOICE \* > from the peak or truck at the beginning of the music and the lowering so regulated as to < \* ABOUT \* > be completed < \* PASSIVE VOICE \* > at the last note. < \* 32 \* >

< \* TOO MANY ( 6) PREPOSITIONS \* > 1 It is the responsibility of transferring activities to ensure < \* MAKE SURE \* > that members [acquire]

obligated < \* BOUND / COMPELLED \* > < \* PASSIVE VOICE \* > service necessary for tour length selected by the member. < \* 23 \* > < \* TOO MANY ( 5) PREPOSITIONS \* > If, at the time of transfer, a member is [undecided] as to < \* ABOUT \* > which tour [(accompanied/all] others) [he/she] desires, < \* WANTS \* >

sufficient < \* ENOUGH \* > obligated < \* BOUND / COMPELLED \* > service to complete the "all others" tour for the area regardless of the member's pay

# Technical Report 86-001

PAGE 4

grade or length of service, personnel ordered to an overseas activity  
[land-based] or afloat) shall have sufficient < \* ENOUGH \* >  
obligated < \* BOUND / COMPELLED \* > service to ensure < \* MAKE SURE \* >  
completion of the appropriate < \* PROPER / RIGHT \* >  
accompanied < \* WENT WITH \* > tour prescribed for the area to which ordered  
prior to < \* BEFORE \* > execution of < \* EXECUTING \* > orders. < \* 74 \* >  
< \* TOO MANY ( 16) PREPOSITIONS \* > [Attainment] of  
obligated < \* BOUND / COMPELLED \* > service through [actual]  
[reenlistment/extension] of enlistment is required. < \* NEEDED \* >  
< \* PASSIVE VOICE \* > < \* TOO MANY ( 3) PREPOSITIONS \* > Page 13 entry for  
[attainment] of obligated < \* BOUND / COMPELLED \* > service is prohibited.  
< \* PASSIVE VOICE \* > Personnel approaching fleet [reserve/retirement]  
eligibility. Personnel who have completed 15 or more years active duty for  
the period of time required < \* NEEDED \* > to complete the applicable tour  
prescribed for the overseas area to which assigned. < \* 29 \* >  
< \* TOO MANY ( 6) PREPOSITIONS \* > Transfer to the Fleet Reserve or Retired  
List will not normally < \* TRY TO AVOID \* > be authorized < \* PASSIVE VOICE \* >  
prior to < \* BEFORE \* > completion of the prescribed overseas tour. Personnel  
approaching retirement eligibility are cautioned < \* PASSIVE VOICE \* > of the  
fact that the requirement to complete the applicable tour for the overseas  
area may [disallow] their eligibility to request < \* ASK \* > a [twilight]  
tour. < \* 30 \* > Personnel (other than CT, OT, RM, and enlisted women in those  
ratings listed in Article 3.181) will not normally < \* TRY TO AVOID \* > be  
[involuntarily] required < \* NEEDED \* > < \* PASSIVE VOICE \* > to serve  
consecutive overseas tours unless Navy [requirements] (billet [priorities])  
direct such action. < \* 35 \* > Personnel assigned consecutive overseas tours

Technical Report 86-001

PAGE 5

as a result of < \* BECAUSE / EVEN THOUGH > [voluntary] action will be  
required < \* NEEDED > < \* PASSIVE VOICE > to serve the  
appropriate < \* PROPER / RIGHT > [accompanied/all] others tour length.  
< \* PARAGRAPH TOO LONG ( 33 SENTENCES WITH 881 WORDS ) >  
< \* 3.0% PERSONAL SENTENCES >

Technical Report 86-001

FINAL ANALYSIS

-----  
READABILITY RESULTS  
-----

Number of Sentences  
33

Number of Words  
881

Number of Syllables  
1509

Avg. Number of Words per Sentence  
26.69

Avg. Number of Syllables per Word  
1.71

Percent of Uncommon Words  
3.06

Percent of Long Sentences  
60.60

GRADE LEVEL (Based on DoD Readability Standard) = 15.03

# Technical Report 86-001

UNCOMMON WORDS			
WORD	FREQ	WORD	FREQ
Attainment	1	involuntarily	1
accompanied/all	2	land-based	1
acquire	1	medley	1
actual	1	paraded	1
alterations	1	priorities	1
attainment	1	reenlistment/extensi	1
beginnning	1	requiremnts	1
ceremonial	1	reserve/retirement	1
courtesy	1	respectively	1
disallow	1	twilight	1
entirety	1	undecided	1
formally	1	voluntary	1
he/she	1	warrant	1

# Technical Report 86-001

UNCOMMON WORDS			
WORD	FREQ	WORD	FREQ
Attainment	1	involuntarily	1
accompanied/all	2	land-based	1
acquire	1	medley	1
actual	1	paraded	1
alterations	1	priorities	1
attainment	1	reenlistment/extensi	1
beginning	1	requiremnts	1
ceremonial	1	reserve/retirement	1
courtesy	1	respectively	1
disallow	1	twilight	1
entirety	1	undecided	1
formally	1	voluntary	1
he/she	1	warrant	1

Technical Report 86-001

DISTRIBUTION LIST

DOD

ASD (MRA&L (Mr. R. Shores))  
USD/R&E (CDR P. Chatelier)

Navy

ASSTSECNAV (SL (Mr. Paschke), MRA)  
CNO (OP-11, OP-01B7, OP-0987H, OP-914D, OP-987, OP-115, OP-12, 40B (Mr. Cale),  
401E (CDR E. Strasser))  
CNR (270, 442 (3 copies))  
CNET (01, 00A, 00A21, N-912, L02)  
CNAVRES (U2)  
COMNAVAIRSYSCOM (03, 340F, 413E, Code 310I (D. Hurst), Code 4003 (D. Kyle))  
NAVEDTRAPRODEVCECEN (PDM, Technical Library (2 copies))  
CNTECHTRA (N-6 (2 copies))  
CO FLEASWTRACENLANT  
CO FLEASWTRACENPAC  
CNATRA (Library (2 copies))  
COMTRALANT (00, Education Advisor)  
COMTRAPAC (2 copies)  
NAVSHIPYD Charleston  
NAVSHIPYD Pearl Harbor (Code 302.3)  
SHIPYDTRSUPPCENLANT (Code 1400)  
NAVPEKSRANDCEN (Library (3 copies), Code 260 (Dr. R. Blanchard),  
Code 71 (Dr. R. Smillie))  
NAVPGSCOL (Code 32, 2124, Library)  
CO FCTCLANT  
CO FCTCPAC  
CO NAVEDTRASUPPCENPAC (2 copies)  
CO NAVEDTRASUPPCENLANT (N-3 (2 copies))  
CO NAMTRAGRU  
CO NAVTECHTRACEN Corry Station (Cryptologic Training Department,  
101B (2 copies), 3330)  
CO NAVTECHTRACEN Meridian (00, Code 700)  
CO NAVTECHTRACEN Treasure Island (Technical Library)  
CO TRITRAFAC Bangor (2 copies)  
CO FLETRACEN Mayport  
CO FLETRACEN San Diego  
CO FLETRACEN Norfolk  
USNA (Chairman, Behavioral Science Department; Library)  
NAVAEROSPMEDRSCHLAB (Aviation Psychology Division)  
NAVAEROSPMEDINST (Code 11, 13)  
CO NAVSUBTRACENPAC (2 copies)  
CO NAVSUBSCOL NLON (Code 0110)  
DIR NAVEDTRAPRODEVCECENDET Memphis  
DIR NAVEDTRAPRODEVCECENDET Great Lakes  
CO NAVEDTRAPRODEVCECEN (PDM, Technical Library (2 copies), IPD)

Technical Report 86-001

DISTRIBUTION LIST (Continued)

COMFLETRAGRU Pearl Harbor  
COMNAVSWC (Code E411)  
CO SERVSCOLCOM Great Lakes (CISO, 700)  
CO NAVEDTRAPRODEVCEEN (PDM, Technical Library (2 copies), IPD)  
COMFLETRAGRU Pearl Harbor  
COMNAVSWC (Code E411)  
CO SERVSCOLCOM Great Lakes (CISO, 700)  
CO SERVSCOLCOM San Diego (Naval Contract Program)  
NATTC Memphis (CISO, Code 011)  
CO FLEMINIEWARTRACEN  
CO HSL-31 (Training Department)  
OIC NAVSPSERVADMINACT (Head, Training Support Unit)  
CO NAVAVSCOLSCOM (0212)  
COMNAVSEASYSYSCOM (SEA-05G (CAPT L. Jones, Mr. D. Waters))  
COMNAVSUPSYSCOM (PML-550 (Mr. W. Gorham))  
NPPS (Mr. J. Karpovich)  
COMNAVELEXSYSCOM (Code 8122, Mr. J. Sizer)  
NAVAIRTECHSERVFAC (Code 00/01, Code 01 (A. Turetsky), Code 122  
(A. Abramson/T. Martin))  
NAVTRASYSCEN (TIC, 001, 002, 1, 7, 71 (2 copies), 10 (3 copies))  
NAVSHIPWPNSYSENGSTA (Code 5700 (Mr. G. Hibner), Code 5710 (Mr. N. Goldfarb,  
Mr. L. Thomson))  
NAVWARCOL (Library)  
DTNSRUC (Code 1803, 5211.1, 522, 522.2, 93)

Air Force

USAFOMC/UMT  
Hq ATC/TTXA  
AFMTC/XR, Lackland Air Force Base  
Air Force Office of Scientific Research/NL, Bolling Air Force Base  
Air Force Office of Scientific Research/NL, AEGIS CSEDS, Moorestown, NJ  
4235 STS/DUTE, Carswell Air Force Base  
CNET Liaison Officer, AFHRL Operations Training Division, Williams Air Force  
Base  
AFHRL (Mr. R. Johnson, Mr. D. Gunning)  
Headquarters Tactical Air Command (DOOS), Langley Air Force Base  
Hq USAF/SEGE (Mr. W. Stiegmann)  
Hq AFLC/LOLMP

Army

Commandant, TRADOC (Technical Library)  
ARI (PERI-IC, PERI-SM, PERI-IF (Dr. J. P. Kincaid, Dr. H. Ozkaptan),  
Library (2 copies), Mr. R. Wisher, Reference Service)  
ARI Field Unit, Fort Knox (PERI-IK)  
ARI Field Unit, Leavenworth

Technical Report 86-001

DISTRIBUTION LIST (Continued)

Army (continued)

USA Materiel Development and Readiness Command (DRCSM-PMP/Mr. D. Norris)  
COMDT, USAIPRM (ATSG-DT-R)  
COM USA Armament Materiel Readiness Command (DRSAR-MAS)  
USA DARCOM (Mr. A. Rulon)  
Army Communicative Technology Office (Col. R. Bernd)  
PM TRADE (Dr. R. Hofer, Mr. W. Stansberry)

Coast Guard

COMDT COGARD (G-P-1/2/42, GRT/54)

Marine Corps

CMC (OT)  
CG MCDEC (Education Advisor)  
CO MARCORCOMMELECSOL  
MCI

Other

CNA  
DLA  
IDA (Dr. J. Orlansky)  
Defense Materiel Specifications and Standards Office (Mr. J. Richardson)  
DARPA (Mr. C. Kelley)

Information Exchanges

DTIC (12 copies)  
ERIC (2 copies)  
DLSIE  
APA (Psychological Abstracts)

END

DTIC

7-86